INTERNATIONAL STANDARD

ISO 13606-1

Second edition
2019-06

Health informatics — Electronic health record communication —

Part 1: Reference model

Informatique de santé — Communication du dossier de santé informatisé —

Partie 1: Modèle de référence

Reference number
ISO 13606-1:2019(E)

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Fax: +41 22 749 09 47
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 215, *Health Informatics*.

This second edition cancels and replaces the first edition (ISO 13606-1:2008), which has been technically revised. The main changes compared to the previous edition are summarised in the Introduction.

A list of all parts in the ISO 13606 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

0.1 Preface

The overall goal of this document is to define a rigorous and stable information architecture for communicating part or all of the Electronic Health Record (EHR) of a single subject of care (patient), or for a group of patients whose information might need to be communicated together (for example, a family). This is to support the interoperability of systems (see Annex C). and components that need to communicate (access, transfer, add or modify) EHR data:

— preserving the original clinical meaning intended by the author;

— incorporating the necessary provenance metadata to inform the recipient or receiving system about the context in which the EHR data were obtained and composed;

— observing and communicating the confidentiality of that data as intended by the author and subject of care.

This document considers the EHR to be the persistent longitudinal and potentially multi-organisation or multi-national record of health and care provision, most often relating to a single subject of care (the patient), created and stored in one or more physical systems in order to inform each subject's future healthcare and to provide a medico-legal record of care that has been provided. This corresponds to the definition provided in ISO 18308:2011 (Requirements for an Electronic Health Record Architecture).

This document is not intended to specify the internal architecture or database design of EHR systems or components, nor is it intended to prescribe the kinds of clinical applications that might request or contribute EHR data in particular settings, domains or specialities. For this reason, the information model proposed here is called the EHR Extract, and might be used to define a message, an XML document or schema, or an object interface. These might be used to communicate EHR data between two repositories, to update a centralised regional or national EHR repository, or within a distributed network of EHR components, systems and services. Whilst an EHR service or system will need to interact with many other services or systems providing terminology, medical knowledge, guidelines, workflow, security, persons registries, billing etc. this document has only touched on those areas if some persistent trace of such interactions is required in the EHR itself, and requires specific features in the reference model to allow their communication.

This document may offer a practical and useful contribution to the design of EHR systems but will primarily be realised as a common set of external interfaces or messages built on otherwise heterogeneous clinical systems. The components that might support an interface conforming to this document will be not only electronic health record systems but also other middleware services such as security components, guideline and workflow systems, alerting and decision support services, personal health systems and applications, sensors and wearable devices, and medical knowledge management services. This document might also prove useful for communicating data about individuals between electronic health record systems and population registries, and also for conducting (approved) research using electronic health records.

This document is part of a five-part standard series, published jointly by CEN and ISO through the Vienna Agreement.

In this document dependency upon any of the other parts of this series is explicitly stated where it applies.

0.2 Technical approach

This document is the second version of an original standard which was published in 2007 by CEN, and in 2008 by ISO. This revision has taken into account the experiences gained by EHR system developers and by large scale eHealth programmes from using the original standard. These were ascertained through an international survey, a wide range of 1:1 interviews, a review of the academic literature, and interactions with many experts active in R&D relating to the EHR. It also meets the relevant requirements in ISO 18308:2011 (Requirements for an Electronic Health Record Architecture). The

revision has taken into account, and aligns as far as possible, with other CEN and ISO Standards and Technical Specifications with which this document might also be used, with international terminology standards and with emerging standards from HL7: Fast Healthcare Interoperability Resources (FHIR). The specifications in this document have drawn from, and align as far as possible with, the reference model specifications published by the openEHR Foundation, and with the archetype models published by the openEHR Foundation and by the Clinical Information Modeling Initiative (CIMI).

The information model in this document is an Information Viewpoint of the ISO Reference Model for Open Distributed Processing (ISO/IEC 10746-1:1998**)**.

Given the diversity of deployed EHR systems, this document has made most features of EHR communication optional rather than mandatory. However, some degree of prescription is required to make EHR Extracts safely processable by an EHR recipient system, which is reflected through mandatory properties within the models in Parts 1, 2, and 4 of this series, and through normative term lists (defined in Part 3 of this series).

0.3 The Dual Model approach

The challenge for EHR interoperability is to devise a generalised approach to representing every conceivable kind of health record data structure in a consistent way. This needs to cater for records arising from any profession, speciality or service, whilst recognising that the healthcare data sets, value sets, templates etc. required by different healthcare domains will be diverse, complex and will change frequently as clinical practice and medical knowledge advance. This requirement is part of the widely acknowledged health informatics challenge of semantic interoperability.

The approach adopted by this standard series distinguishes a Reference Model, defined in this document and used to represent the generic properties of health record information, and Archetypes (conforming to an Archetype Model, defined in Part 2 of this series), which are meta-data used to define patterns for the specific characteristics of the healthcare data that represents the requirements of each particular profession, speciality or service.

The Reference Model represents the global characteristics of health record components, how they are aggregated, and the context information required to meet ethical, legal and provenance requirements. This model defines the set of classes that form the generic building blocks of the EHR. It reflects the stable characteristics of an electronic health record, and would be embedded in a distributed (federated) EHR environment as specific messages or interfaces (as specified in Part 5 of this series).

This generic information model needs to be complemented by a formal method of communicating and sharing the organisational structure of predefined classes of EHR fragment corresponding to sets of record components made in particular clinical situations. These are effectively pre-coordinated combinations of named RECORD_COMPONENT hierarchies that are agreed within a community in order to ensure interoperability, data consistency and data quality.

An Archetype is the formal definition of prescribed combinations of the building-block classes defined in the Reference Model for particular clinical domains or organisations. An archetype is a formal expression of a distinct, domain-level concept, expressed in the form of constraints on data whose instances conform to the reference model. For an EHR_EXTRACT, as defined in this document, an archetype instance specifies (and effectively constrains) a particular hierarchy of RECORD_COMPONENT sub-classes, defining or constraining their names and other relevant attribute values, optionality and multiplicity at any point in the hierarchy, the data types and value ranges that ELEMENT data values may take, and other constraints.

This document recognises that archetypes (or equivalent clinical models) are not always directly incorporated within the present-day architectures of electronic health record systems. This document therefore does not mandate that archetypes are used within such systems. It does, however, require that the clinical information models or equivalents (data items, data item aggregations, data value constraints, terminology bindings, units of measure etc.) that have been used to generate an EHR_EXTRACT are themselves created and communicated, or referenced, within each EHR_EXTRACT. These communicated or referenced archetypes have to conform to Part 2 of this standard series, and maybe communicated through an interface conforming to part 5 of this Standard series.

0.4 Overview of the EHR_EXTRACT record hierarchy

The information in a health record is inherently hierarchical. Clinical observations, reasoning and intentions can have a simple or a more complex structure. They are generally organised under headings, and contained in "documents" such as consultation notes, letters and reports. These documents are usually filed in folders, and a subject of care may have more than one folder within a healthcare enterprise (e.g. medical, nursing, and obstetric).

The EHR Communications Reference Model needs to reflect this hierarchical structure and organisation, meeting published requirements in order to be faithful to the original clinical context and to ensure meaning is preserved when records are communicated between heterogeneous clinical systems. To do this, the model formally sub-divides the EHR hierarchy into parts that have been found to provide a consistent mapping to the ways which individual EHRs are organised within heterogeneous EHR systems.

These parts are summarised in Table 1 below.

Table 1 — Main hierarchy components of the EHR Extract Reference Model

EHR HIERARCHY COMPONENT	DESCRIPTION	EXAMPLES
EHR_EXTRACT	The top-level container of part or all of the EHR of a single subject of care or for a group of subjects of care (such as a family), for communication between an EHR Provider system and an EHR Recipient.	(Not applicable)
FOLDER	The high level organisation within an EHR, dividing it into compartments relating to care provided to a single subject of care, for a single condition, by a clinical team or institution, or over a fixed time period such as an episode of care.	Diabetes care, Schizophrenia, Cholecystectomy, Paediatrics, St Mungo's Hospital, GP Folder, Episodes 2000-2001.
COMPOSITION	The set of information committed to one EHR by one agent, as a result of a single clinical encounter or record documentation session.	Progress note, Laboratory test result form, Radiology report, Referral letter, Clinic visit, Clinic letter, Discharge summary, Functional health assessment, Diabetes review.
SECTION	EHR data within a COMPOSITION that belongs under one clinical heading, usually reflecting the flow of information gathering during a clinical encounter, or structured for the benefit of future human readership.	Reason for encounter, Past history, Family history, Allergy information, Subjective symptoms, Objective findings, Analysis, Plan, Treatment, Diet, Posture, Abdominal examination, Retinal examination.
ENTRY	The information recorded in an EHR as a result of one clinical action, one observation, one clinical interpretation, or an intention. This is also known as a clinical statement.	A symptom, an observation, one test result, a prescribed drug, an allergy reaction, a diagnosis, a differential diagnosis, a differential white cell count, blood pressure measurement.
CLUSTER	The means of organising nested multi-part data structures such as time series, and to represent the columns of a table.	Audiogram results, electro-encephalogram interpretation, weighted differential diagnoses.
ELEMENT	The leaf node of the EHR hierarchy, containing a single data value.	Systolic blood pressure, heart rate, drug name, symptom, body weight.

An EHR_EXTRACT contains EHR data as COMPOSITIONs, organised in a FOLDER hierarchy.

COMPOSITIONs contain ENTRYs, optionally contained within a SECTION hierarchy.

ENTRYs contain ELEMENTS, optionally contained within a CLUSTER hierarchy.

Representing participation: The Reference Model in the previous version of this standard provided explicit classes at certain parts of the Record Component hierarchy through which it was possible to represent the identity and roles played by actors contributing to healthcare and to its documentation. In this version of the Reference Model the LINK class is intended to be used to reference demographic entities. The roles played by these entities can be labelled using extended term lists defined in Part 3 of this standard series. This updated mechanism offers greater flexibility than the previous version in where the references to such democratic entities may be represented within the Record Component hierarchy.

Representing context: Any EHR_EXTRACT references any other RECORD_COMPONENTS that are connected to the communicated content, for example via the RECORD_COMPONENT hierarchy and via LINK targets. If the EHR exchange service (e.g. as specified in Part 5) permits access to referenced components, any user would be able to access and review any additional areas of content that were not originally included. (Archetypes bring together the key elements of immediate documentation context.)

Representing authenticity: Every EHR_EXTRACT may contain attested views: these might be PDF or html or other renderings that are the authentic view of what was seen and persisted by the original author. The proof may also optionally be included, which is the evidence of a digital signature.

EHR_EXTRACTS are created for specific purposes, and will not automatically guarantee that these will be fit for other purposes.

0.5 Summary of changes made in this edition of the standard

The scope of all parts remains the same.

The objective of this revision was to:

— obtain implementer feedback on adoption experiences with the published version of the 13606 standard series;

— simplify the reference model by removing properties that have not proved useful to implementers;

— improve the demographics model to support the use of demographic archetypes;

— improve alignment with ISO 13940 System of concepts to support continuity of care (Contsys);

— align the data types with ISO 21090 Harmonized data types for information interchange (see Annex A);

— prepare the ground for alignment with HL7 FHIR;

— update the archetype model to align with the openEHR Archetype Object Model 2.0;

— include reference archetypes for commonly needed information (e.g. demographics);

— update the audit log model to align with ISO 27789 Audit trails for electronic health records, and the ISO 22600 series, Privilege management and access control.

Reference Model changes

Base Component

A class Base Component has been introduced higher in the inheritance hierarchy than Record Component, which has a unique identifier, version history information and attestation information.

This allows all of the structures within an EHR Extract to be version managed and attested, including LINK and demographic information, as well as the original family of Record Components.

Record Component

Several properties that had not proved useful have been removed from Record Component.

Importantly, the model now semantically labels Record Components through archetype ID, avoiding duplicating and possibly conflicting semantic labels such as name and meaning.

Experience is that these different properties were not differentially well used, and resulted in inconsistent practices.

Consultation with vendors and providers who do not intend to use archetypes has confirmed that they could create a library of archetypes mapping their data structures, should they choose to adopt this standard series.

Properties relating to sensitivity and policy ID have been moved to Composition, to avoid the risk of a Composition containing data of mixed policies and therefore inconsistently complete access by different parties.

Structure Component

A generic parent class Structure Component is now the universal parent class of all Record Components and demographic classes.

All such classes inherit an archetype ID, which now also importantly allows demographic structures to be defined through archetypes, which was a popular change request.

EHR_Extract

Extract Criteria has been removed as implementers did not find it useful.

EHR_EXTRACT may now contain a set of extracted EHR components, and so may contain data on multiple subjects of care.

Folder

The Folder has the property subject of care, which allows an EHR extract potentially to contain information about more than one subject of care, such as a family, which was an important change request we received.

The EHR Extract is a kind of folder, with specific meta data about the extract generation.

Composition, Entry and Cluster

Some unused properties have been removed from Composition, Entry and Cluster.

This includes session_time, obs_time. Such dates and times are better included within relevant archetypes with more precisely specified names and roles.

Element

The Element class and its counterpart Demographic Element, are more genuinely leaf nodes with fewer inherited properties and fewer inherited associations than in the past.

This is a response to a number of ICT vendors who indicated that the original Element was too property rich, inviting inconsistent adoption practice.

Null flavour is no longer an explicit Reference Model property. Instead a Reference Archetype has been defined in Part 3, to allow null to be asserted not only at Element level but higher up the hierarchy (e.g. that a Cluster or Entry is not present).

Data Value

The data values are now all represented as a constrained subset of the data types in ISO 21090, conforming to its mechanism for profiling (see Annex A).

Demographics

Rather than being a separate package, demographic entities are represented using classes that inherit many of the same mechanics as Record Components, simplifying adoption.

This is now also means that demographic entries in a demographic extract can be uniquely identified, version managed and attested.

It is now much easier to refer specifically to actors within roles at care settings, in cases where actors play multiple roles and work within different care settings over time, which is relatively common.

Link

The Link class has been simplified, but enriched so that Links have a unique identifier, and can be versioned and attested.

Links can connect any identified components including demographic entities, which is now the way that most participations are represented. This was an important and well supported change request, to correct a less successfully modelled part of the previous standard.

The vocabulary for LINK has been greatly extended within part 3, and where appropriate aligned with Contsys.

LINKS can continue to be used to represent clinically relevant connections between parts of a record, point to point or as a linkage thread.

The former class FUNCTIONAL_ROLE and RELATED_PARTY have been removed, as the connection to demographic entities can now be made through archetyped DEMOGRAPHIC ENTITY instances connected to the relevant RECORD_COMPONENT via LINK.

A new class EXTERNAL_LINK enables references to be included to non-EHR data such as care protocols, safety reports or academic publications.

Audit information

The audit info class now aligns with ISO 27789, as does the corresponding audit model in Part 4 of this standard series.

0.6 Relationship of this standard to other relevant standards

— The data types used in this document are a profile of ISO 21090 Health informatics - Harmonized data types for information interchange.

— Alignment has especially been undertaken with ISO 13940 Health informatics - System of concepts to support continuity of care (Contsys). All of the terms and definitions within the standard series have been harmonised across the five parts, with most of the terms being in Part 1. All of them have been aligned with Contsys.

— An important 13606 FHIR profile project is in progress within HL7 (see Annex B). No challenges have been identified with being able to create such a profile. However there are a few areas of mapping alignment which need further work between ISO 13606 and HL7.

— Parts 1 and 4 align with ISO 27789 Health informatics - Audit trails for electronic health records. Alignment in Part 4 has been maintained with ISO 22600 Health informatics - Privilege management and access control.

Health informatics — Electronic health record communication —

Part 1: Reference model

1 Scope

This document specifies a means for communicating part or all of the electronic health record (EHR) of one or more identified subjects of care between EHR systems, or between EHR systems and a centralised EHR data repository.

It can also be used for EHR communication between an EHR system or repository and clinical applications or middleware components (such as decision support components), or personal health applications and devices, that need to access or provide EHR data, or as the representation of EHR data within a distributed (federated) record system.

This document will predominantly be used to support the direct care given to identifiable individuals or self-care by individuals themselves, or to support population monitoring systems such as disease registries and public health surveillance. Uses of health records for other purposes such as teaching, clinical audit, administration and reporting, service management, research and epidemiology, which often require anonymization or aggregation of individual records, are not the focus of this document but such secondary uses might also find the document useful.

This Part 1 of the multipart series is an Information Viewpoint specification as defined by the Open Distributed Processing – Reference model: Overview (ISO/IEC 10746-1). This document is not intended to specify the internal architecture or database design of EHR systems.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 21090:2011, *Health informatics — Harmonized data types for information interchange*

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1 Actors

3.1.1
attester
actor (person) who certifies and records legal responsibility for a particular unit of information

3.1.2
committer
agent (party, device or software) whose direct actions have resulted in information being committed

3.1.3
composer
healthcare actor responsible for creating, synthesising or organising information that is committed to an electronic health record.

Note 1 to entry: This agent takes responsibility for its inclusion in that electronic health record, even if not the originator of it and even if not the committer of it.

3.1.4
electronic health record provider
healthcare actor in legitimate possession of electronic health record data and in a position to communicate it to another appropriate entity

3.1.5
healthcare provider
care provider
health provider
health service provider
healthcare service provider
healthcare actor that is able to be assigned one or more care period mandates

Note 1 to entry: The personnel of a *healthcare organization* that is a *healthcare provider* may include both *healthcare professionals* and others which participate in the provision of *healthcare*.

Note 2 to entry: This document includes only two specializations of *healthcare provider*. This is not meant to exclude the possibility of other specializations. In jurisdictions where other kinds of *healthcare actors* are included in the concept of *healthcare provider*, the necessary specializations may be added.

Note 3 to entry: According to this definition, *organizations* solely responsible for the funding, payment, or reimbursement of *healthcare* provision are not *healthcare providers*; for the purpose of this document they are considered as *healthcare third parties*.

[SOURCE: ISO 13940:2015]

3.1.6
role
function or position

[SOURCE: ISO/HL7 21731:2014]

3.1.7
subject of care
subject of healthcare
patient, client
service user
healthcare actor with a person role; who seeks to receive, is receiving, or has received healthcare

Note 1 to entry: A fœtus may be considered as a *subject of care* when receiving *healthcare*.

EXAMPLE A treated patient, a client of a physiotherapist, each particular member of a target population for screening, each particular member of a group of diabetic people attending a session of medical education, a *person* seeking health advice.

[SOURCE: ISO 13940:2015]

3.2 Concepts and terms

3.2.1
generic
applicable to requirements or information models across healthcare professions, domains and countries

3.3 Information management

3.3.1
abstract class
class that cannot be instantiated independently

[SOURCE: ISO/IEC/IEEE 24765:2017, 3.24]

3.3.2
archetype
instance of an archetype model, specifying the clinical concept and the value constraints that apply to one class of Record Component instances in an electronic health record extract

3.3.3
archetype model
information model of the metadata to represent the domain-specific characteristics of electronic health record entries, by specifying values or value constraints for classes and attributes in the electronic health record reference model

3.3.4
client application
healthcare application which is behaving at that moment as a requester of health record data from a shareable electronic health record

3.3.5
data
reinterpretable representation of *information* (3.3.9) in a formalized manner suitable for communication, interpretation or processing

Note 1 to entry: Data are defined without any context in such a way that by itself one cannot tell its correct meaning, where it has meaning this is information.

[SOURCE: ISO/IEC 2382 2015, 2121272]

3.3.6
data repository
an identifiable *data* (3.3.5) storage facility

Note 1 to entry: In ISO 10303 22:1998 this is the definition of repository.

[SOURCE: ISO 13940:2015]

3.3.7
healthcare data
data (3.3.5) produced during healthcare activities

[SOURCE: ISO 13940:2015]

3.3.8
healthcare information
information (3.3.9) about a person, relevant to his or her healthcare

[SOURCE: ISO 13940:2015]

3.3.9
information
knowledge concerning objects that within a certain context has a particular meaning

Note 1 to entry: Facts, events, things, processes, and ideas, including concepts, are examples of objects.

Note 2 to entry: Information is something that is meaningful. Data might be regarded as information once its meaning is revealed.

[SOURCE: ISO 13940:2015]

3.3.10
distributed processing
information (3.3.9) processing in which discrete components may be located in different places, and where communication between components may suffer delay or may fail

[SOURCE: ISO/IEC 10746-2:2009]

3.3.11
health record
data repository regarding the health and healthcare of a subject of care

Note 1 to entry: The term *electronic health record* may be used for a *health record* where all *information* is stored on electronic media. However, this concept is not formally defined in this document.

Note 2 to entry: A *health record* may include, for example, medical records, dental records, social care records.

[SOURCE: ISO 13940:2015]

3.3.12
health record component
part of a *health record* (3.3.11) that is identifiable for the purposes of referencing and revision

Note 1 to entry: This document defines one *health record component* specialization, the *electronic record component*. However, as the content of a *health record* is not limited to *information* in electronic format, the content of *health record components* may be in formats other than electronic.

Note 2 to entry: A *health record component* may itself result from an aggregation of multiple *health record components*.

[SOURCE: ISO 13940:2015]

3.3.13
electronic health record
health record (3.3.11) where all information is stored on electronic media

3.3.14
electronic health record system
system for recording, retrieving and manipulating information in *electronic health record* (3.3.13)

[SOURCE: ISO 18308:2011]

3.3.15
health record extract
part or all of a *health record* (3.3.11) extracted for the purpose of communication

[SOURCE: ISO 13940:2015]

3.3.16
healthcare application
system for collecting, saving, processing, and presenting *healthcare information* (3.3.8) by means of a computer.

Note 1 to entry: This definition is based on the definition of application in ISO/IEC 19770-5:2015.

3.3.17
information architecture
structural design of an information space to facilitate task completion and intuitive access to content

3.3.18
metadata
data (3.3.5) that defines and describes other data

[SOURCE: ISO/IEC 11179-3:2013, 3.2.74]

3.3.19
semantic interoperability
ability for data shared by systems to be understood at the level of fully defined domain concepts

[SOURCE: ISO 18308:2011]

3.3.20
view
alternate presentation of *data* (3.3.5) for a different purpose

3.4 Privacy and security

3.4.1
access control
means of ensuring that the resources of a *data* (3.3.5) processing system can be accessed only by authorised entities in authorised ways

[SOURCE: ISO 17090-1:2013]

3.4.2
audit trail
collection of audit records from one or more audit logs relating to a specific subject of care or a specific *electronic health record* (3.3.13)

[SOURCE: ISO 27789:2013]

3.4.3
authentication
provision of assurance of the claimed identity of an entity by securely associating an identifier and its authenticator

[SOURCE: ISO 22600-3:2014]

3.4.4
authorisation
granting of privileges

[SOURCE: ISO 22600-1:2014]

3.4.5
confidentiality
property that *information* (3.3.9) is not made available or disclosed to unauthorised individuals, entities, or processes

[SOURCE: ISO 7498-2:1989]

3.4.6
digital signature
data (3.3.5) appended to, or a cryptographic transformation of, a data unit that allows a recipient of the data unit to prove the source and integrity of the unit and protect against forgery e.g. by the recipient

[SOURCE: ISO 7498-2:1989]

3.4.7
privacy
freedom from intrusion into the private life or affairs of an individual when that intrusion results from undue or illegal gathering and use of *data* (3.3.5) about that individual

[SOURCE: ISO/IEC 2382-8:2015, 2126263]

3.5 Process management

3.5.1
attest
activity of certifying and recording legal responsibility for a particular unit of *information* (3.3.9)

3.5.2
healthcare process
care process
set of interrelated or interacting healthcare activities which transforms inputs into outputs

Note 1 to entry: The main type of healthcare process is the clinical process that has a health state as input and output and includes all activities in relation to one or more specified health issues.

Note 2 to entry: A healthcare process is not by definition restricted to one healthcare provider or any other organizational unit borders.

[SOURCE: ISO 13940:2015]

4 Abbreviations

For the purposes of this document, the following abbreviations apply.

CEN	Comité Européen de Normalisation (European Committee for Standardization, a federation of 28 national standards bodies that are also ISO member bodies)
EHR	Electronic Health Record
EU	European Union
GP	General Practitioner
HL7	Health Level Seven
IANA	Internet Assigned Numbers Authority
IETF	Internet Engineering Task Force
ISO	International Organization for Standardization
OID	Object Identifier
R&D	Research and Development
RFC	Request For Comments

UCUM	Unified Code for Units of Measure
UML	Unified Modelling Language
XML	Extensible Mark-up Language

5 Overview

5.1 Overview of the reference model

Figure 1 shows the reference model for EHR extracts and demographic extracts.

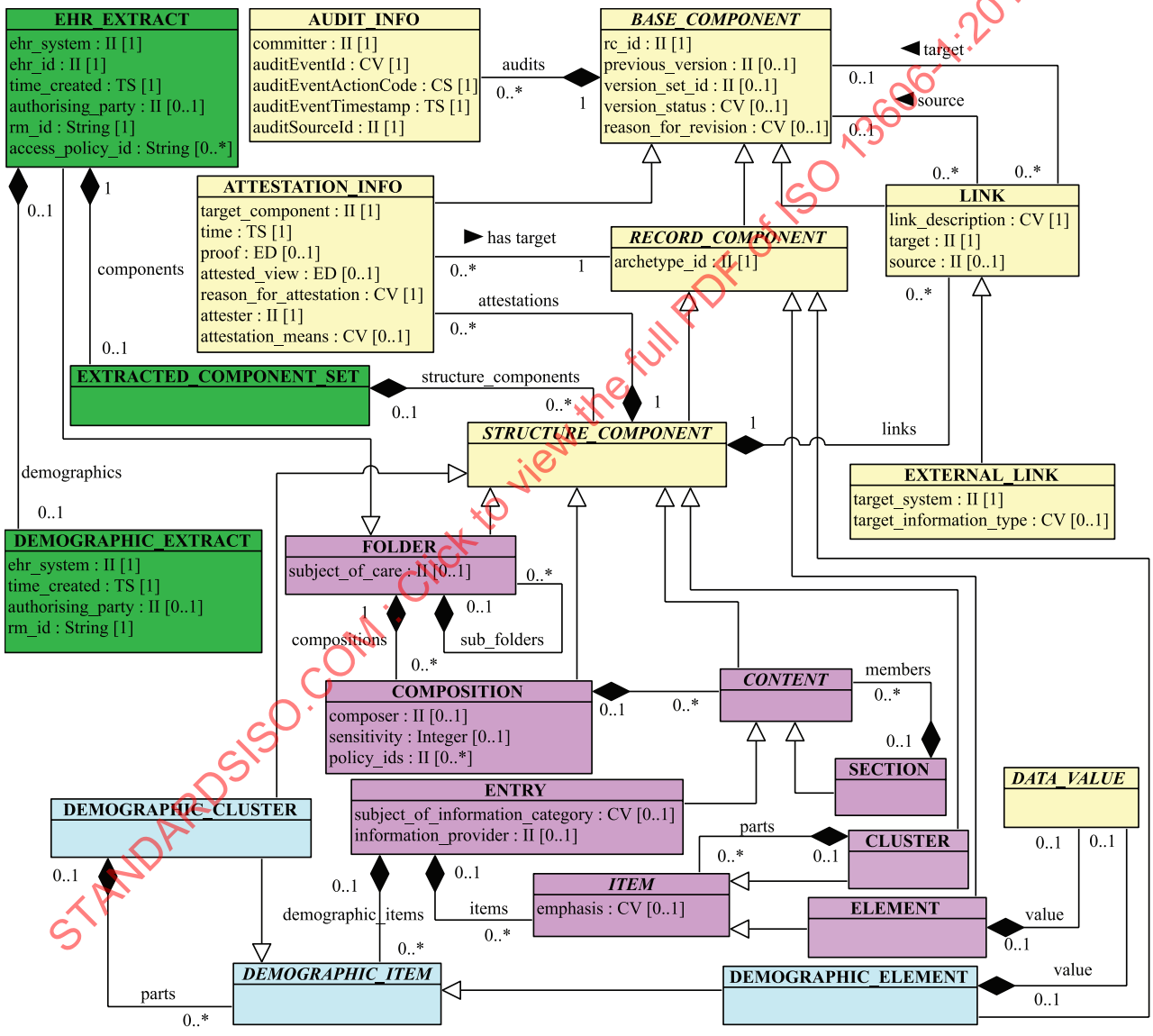

Figure 1 — Reference model for EHR extract and demographics

NOTE The colours in Figure 1, and subsequent diagrams, are offered only as an aid to the reader and have no normative significance. The green coloured classes are the high level containers of the EHR content being communicated. The purple classes represent the main specialisations of RECORD_COMPONENT that provide the main hierarchical organisation of the healthcare information being communicated. The blue classes provide a similar hierarchical organisation to the demographic information being communicated. The cream coloured classes provide additional context to the other classes of the diagram.

5.2 Representing roles and responsibilities within the EHR _EXTRACT

5.2.1 General

One of the key areas of provenance that are communicated as part of a shared electronic health record is information about the actors involved in care provision and its documentation. Performing a care act in a modern health service can involve a large number of actors, with different roles and responsibilities, each of whom might need to be represented in a subject of care's EHR. The approach taken in most generic EHR architectures, including this document, is to differentiate these into three broad categories.

5.2.2 Actors playing a role in the actual healthcare process

This set will usually include a core actor who is the key person relating to the subject of care during that care activity, and a series of collaborating actors who may be providing or supporting parts of the care, are involved in making decisions, are observers, or are present to support or co-represent the subject of care. These actors might not all be present: for example, the policies of a consultant in charge of care may be followed because the subject of care is under his team, even if he is himself not with the subject of care on that occasion. Sometimes actors might be documenting a case review or a care planning negotiation involving one or more professionals but where the subject of care is not present. On the other hand, the subject of care might be the only actor involved in the care process, for illness self-management or prevention activities.

5.2.3 Actors contributing to the process of documenting care within the EHR

This will usually be a subset of those involved in care (and most commonly, the key actor), but might include people who were not part of delivering the care (e.g. a secretary or a transcriptionist) and may (more so in the future) include the subject of care themselves. It is important to recognise that the different actors will sometimes need to complete separate records of the same activity and may also attest them independently.

5.2.4 Actors confirming the validity of the EHR documentation

The paper analogy of attestation, the term used in this document, is the signing of a letter or report. Most commonly the act of signing a document combines two intentions: to confirm that the document is correct (e.g. is a complete record, is free of errors and omissions) and for the signer to confirm that he or she agrees to the content (e.g. to validate a prescription). In most of these situations the status or seniority of the signer is important. Some of the actors described in a care act will not themselves sign the entries describing their contribution to care: much of healthcare works through delegation. For example, the medical record documentation made by a junior doctor on a ward round is rarely reviewed by the consultant and almost never countersigned. Most observations on a paper monitoring chart are not individually signed. With electronic systems this practice might change, but some level of delegation and trust will probably always exist within care teams.

There is therefore a wide range of potential roles and responsibilities that might need to be represented in an EHR, and as patterns of health service evolve these might change in the future. The goal of the EHR_EXTRACT architecture is to permit any number of actors and roles to be defined within a COMPOSITION: either for the whole COMPOSITION or more narrowly for individual ENTRYs.

The approach taken in this document is:

— to specify a small number of roles that need to be unambiguously communicated to ensure safe interpretation of EHR_EXTRACTs by a receiving system, and which are likely to arise frequently;

— to permit other ad hoc participations to be defined by health services, systems or in individual EHR instances at the COMPOSITION or ENTRY level;

— to permit any number of attestations to be added to the EHR, to sign FOLDERs or COMPOSITIONs or only parts of COMPOSITIONs.

The specific roles that have been defined in this Reference Model are discussed below.

5.2.5 Subject of care

In response to requirements received during the revision of this document, the EHR_EXTRACT may contain part or all of the electronic health record of one or more subjects of care. If the EHR of more than one subject of care is included, these shall be separated at the root FOLDER level within the EHR_EXTRACT. The document does not specify the precise rules governing when it is appropriate to combine extracts of more than one subject of care within a single EHR_EXTRACT, but examples of when this might be meaningful will include communicating records of multiple members of a single family or who live at the same address, multiple persons who have shared a common risk or incident, or are being transferred collectively to receive care in another setting.

However, there will be occasions when the FOLDER for a single subject of care still contains information about more than one person. Some commonly occurring examples are listed here.

— Pregnancy: it is common practice for the mother's record to contain the full pregnancy care record including that of her baby or babies until after birth, when any relevant information is copied into the new records of those babies.

— In utero interventions: in some situations, a new record is created well before a baby is born, perhaps if significant healthcare is required. In such situations the new record is created for the foetus as a convenience to permit a separation of data from the mother's record, and in anticipation of a new legal record for the baby. Depending upon the age of the foetus, and the laws pertaining to each country, either the baby or the mother will be the legal data subject, but in any case there is still a single identifiable subject of care for each record.

— Multiple pregnancy with each foetus having its own record: this is often cited as a situation in which health actions might really "belong" to two, conjoint twins or more subjects of care. In these situations it would seem logical that each baby's EHR_EXTRACT contains a copy of the relevant COMPOSITIONs, rather than attempting a complex join between two or more records to reference a single COMPOSITION held in one of these records. More complex cross-linkage arrangements might be made within local EHR systems, permitting users to enter the data once and have it logically added to both records.

— Donated organs: Some test results relating to the donor of an organ may be appropriate to store in the EHR of the person receiving the donated organ – such as the viral status or genetic record of the donor– as the person will from this time on be a genetic mosaic. For this reason, information in the EHR may be classified as pertaining to a "donor".

The subject of care identifier in the root FOLDER will reference a snapshot of demographic information as held by the EHR Provider, to enable the subject of care to be matched to the demographics repository of the EHR Recipient, and/or for the EHR_EXTRACT to be referenced to the individual subject of care even if external demographics services are not available.

5.2.6 Composer

This actor is the person who has actually composed the words, terms, figures and values etc. that are represented in the COMPOSITION. The composer will almost always have played a key role in the information gathering, thinking or enacting aspects of the healthcare being documented. Sometimes, though, he or she might be a junior team member creating a record on behalf of a team.

The Composer attribute therefore represents the party who composed the data in a COMPOSITION, irrespective of who committed it or who attested it. The COMPOSITION will be seen as being primarily attributed to this person. Whether or not the composer is changed when a revision is made is optional, as it will depend upon the extent of the change made and if the revising party is willing and in a position to assume primary responsibility for the revised COMPOSITION as its composer. Applications may use the composer's name to label COMPOSITION data for display purposes. (The role of team members other than the composer can be added for individual ENTRYs as additional demographic_items.)

5.2.7 Committer

In many situations the person who composes the words is not the one who types them in. A common example is dictated letters and reports, which may be typed up by a secretary or transcriptionist. A junior clinical team member might also describe himself as the committer if he is really only acting as the scribe for another (composing) senior team colleague. In some transcription scenarios the typed text is checked by the composer who then commits it to the subject of care's EHR himself. In some scenarios several clinical team members are working in collaboration to deliver a care service; each of these might be able to document (and attest) their own portions of this care in the subject of care's record.

Other situations might arise in which the committer is not responsible for data entry, for example when a measurement device is directly feeding a clinical application. In these situations the information_provider or demographic_items attributes of ENTRY can be used to supplement the set of defined actors.

5.2.8 Subject of information

This attribute is needed to identify the person about whom the information in an ENTRY relates if not the subject of care. The commonest example of this is family history, but information about the subject of care's friend, life partner, sexual partner, employer, child etc. might sometimes be recorded in an EHR. This needs to be unambiguously distinguished from the majority of the EHR information which is about the subject of care.

In some contexts, parties might only be specified precisely if they are registered within the local demographics service AND they have given their consent to be identified in this subject of care's EHR. This will increasingly arise in clinical fields like cancer genetics that manage subjects of care within their family context. The more common situation is where the subject of care is describing the health of others.

The subject_of_information category permits the relationship of that subject to the subject of care to be defined as a coded term, and optionally for more details of that subject to be provided through a LINK to DEMOGRAPHIC_ITEM. This approach will allow archetypes to be reused with different subjects of care, and the unambiguous processing of EHR ENTRYs to distinguish data about the subject of care from data about other parties. It is also possible to let an ENTRY contain a DEMOGRAPHIC_ITEM with information on the subject of information (but then the *subject of information DEMOGRAPHIC_ITEM* would have to be specified in the archetype, just as all other ITEMs).

5.2.9 Information provider

Most of the information documented in an EHR will originate from the subject of care or one of the other participants in the care act. However at times ENTRYs may be added whose data values have been provided by some other party, for example a relative or carer who might be with the subject of care or seeing the subject of care's doctor alone confidentially. Other clinical parties might provide information indirectly (e.g. by phone) to the composer. The information_provider property of ENTRY allows for the formal identification of such parties.

5.3 About the use of datatypes in this document

A constrained subset of ISO 21090:2011 Health informatics — Harmonized datatypes for information interchange is used both in the reference model and for defining the datatypes (DATA_VALUE specializations) to be used when creating archetypes.

Implementations that claim conformance to this document shall not use any of the following ISO 21090:2011 attributes from the classes ANY and HXIT:

— validTimeLow;

— validTimeHigh;

— controlInformationRoot;

— controlInformationExtension;

— nullFlavor;

— flavorId;

— updateMode.

The following datatypes and UML support types specified in ISO 21090:2011, are used in this document.

— II - Implementations that claims conformance to this document shall not use the ISO 21090:2011 II attributes displayable, scope and reliability.

— TS

— CD.CV (In the diagrams shown as CV)

— ED - Implementations that claims conformance to this document shall not use any of the ISO 21090:2011 ED attributes value, xml, compression, charset, language and translation.

— CS

— TEL.URL (In the diagrams shown as URL) - Implementations that claims conformance to this document shall not use any of the ISO 21090:2011 TEL.URL attributes use, capabilities and usablePeriod.

— Boolean

— Binary

— String

— Integer

— Float

— Uid

The DATA_VALUE specialisations are defined in 7.2. These are the datatypes to be used when creating archetypes.

6 Common Properties of Record Components

6.1 Base, record and structure components

6.1.1 General

6.1.1.1 Base component

The main building block classes that are used to construct the EHR_EXTRACT are kinds of BASE_COMPONENT. This class provides a set of generic properties to all of its inheriting classes, including a unique identifier, information about when and by whom that unit of information was committed to the EHR, if the component has been revised, the reason why and providing references to other historic versions.

It is important that each component of an EHR be uniquely and consistently identified across multiple EHR_EXTRACTS, so that references to or between them remaining valid. Examples of such references are semantic links, revision and attestation. The BASE_COMPONENT unique identifier, the rc_id, is one that shall be issued to this EHR component by the EHR system in which it was first committed (its originating EHR system); other holders of this component need to retain this identifier value, and to reuse it whenever this component is subsequently communicated using this document, in order to ensure that identical components within multiple EHR_EXTRACTs are always consistently identified. This identifier

is also to be used when referring to this component from other parts of the reference model, examples when referring to this as their component that has been revised, or if it is the target of a LINK.

The rc_id attribute is of data type Instance Identifier (II), which incorporates an ISO OID; II is considered internationally to be the most appropriate data type for persistent identifiers that are required to be globally unique. It is unlikely that contemporary EHR systems will have existing primary keys or internal identifiers that correspond directly to globally-unique II instances. However, an EHR Provider system that has been issued with an organisational OID might use its internal references to construct unique local extensions to that OID and thereby construct globally-unique rc_id values. Alternatively, it might create completely new rc_ids and retain a record of the mapping of these to each internal identifier, so that any future EHR_EXTRACTS it generates will use consistent rc_id values. It is also unlikely that an EHR Recipient system will be able to use received rc_id values as its internal primary keys for the data, since every database uses a slightly different approach to generating and using such keys. The EHR Recipient might therefore also need to keep a record of the mapping of imported rc_id values to its primary keys, so that future references to those RECORD_COMPONENTS can be appropriately matched, and it can create EHR_EXTRACTs that reapply those rc_id values to the exported data. An alternative approach is for EHR systems to explicitly store the rc_id values along with the EHR data, and treat this as part of the "payload" data and not attempt to use these also as primary keys. It should also be noted that the rc_id does not function as a primary key equivalent within an EHR_EXTRACT i.e. duplicate values of rc_id are permitted if each instance does indeed refer to the same piece of healthcare data within the EHR Provider system.

In some jurisdictions it is a medico-legal requirement to document and to communicate when and by whom EHR data were entered into an EHR system. This information is represented in BASE_COMPONENT as the class AUDIT_INFO which is a profile of ISO 27789:2013 (Audit trails for electronic health records). This information is optional, to allow for the possibility that some data will have been imported from simple legacy systems for which the original committal data are not known. It is deliberately positioned as part of BASE_COMPONENT to also enable LINK or ATTESTATION_INFO to be associated with a different date, time and committer to the STRUCTURE_COMPONENT containing it. Multiple instances of AUDIT_INFO may be included if there is a requirement to communicate information about successive committal of this information within different EHR repositories over time.

BASE_COMPONENT also represents the version status, an optional reason for revision, the identity of the immediately previous version that was the basis of this revision, and an identifier that is common to all versions – so that non-sequential versions made on different EHR systems can still be related to each other. An optional version status attribute indicates if the present version was, at the time of its committal, knowingly a draft (i.e. intended to be replaced in the near future), an update to a previous draft version, a correction of an erroneous former version, or an empty component that is the logical deletion of its predecessor (e.g. if the predecessor was saved in the wrong EHR). If no status is given, it is assumed that this is the definitive (first) version. The code set for this attribute is defined in Part 3 of this series.

BASE_COMPONENT has three inheriting classes: RECORD_COMPONENT, LINK, ATTESTATION_INFO. RECORD_COMPONENT is described below; the other two classes are described later in this subclause.

6.1.1.2 Record component

RECORD_COMPONENT is the abstract root class of all of the major hierarchical components making up the main structure of the healthcare data (FOLDER, COMPOSITION, SECTION, ENTRY, CLUSTER, ELEMENT), and the structures representing demographic entities (DEMOGRAPHIC_FOLDER, DEMOGRAPHIC_ENTITY, DEMOGRAPHIC CLUSTER, DEMOGRAPHIC_ELEMENT). RECORD_COMPONENT contains an archetype_id property, and thereby relates aggregations of RECORD_COMPONENTs to the archetypes to which they correspond on a node by node basis.

6.1.1.3 Structure component

STRUCTURE_COMPONENT is the abstract class that functions as the common parent to all of the RECORD_COMPONENTs except for the leaf classes ELEMENT and DEMOGRAPHIC_ELEMENT. STRUCTURE_COMPONENTs may contain LINKs to any other BASE_COMPONENTs, and maybe attested.

In the previous edition, the class ELEMENT had the ability to contain any number of instances of the LINK class and the ATTESTATION_INFO class. This added a complexity to ELEMENT that seldom was needed and that required functionality that existing EHR systems seldom has.

If a particular use-case requires an ELEMENT that has the ability to contain instances of the LINK class or the ATTESTATION_INFO class, then a cluster containing only that ELEMENT may be used.

6.1.2 Base component

Term: *base component*

Definition: abstract class that include the properties common for all revisable classes included in EHR extract

Class name: BASE_COMPONENT

Description: This is an abstract class that include the properties that are common for all revisable classes in the EHR extracts and in the demographic extract.

A model showing the associations of this class is shown in Figure 2.

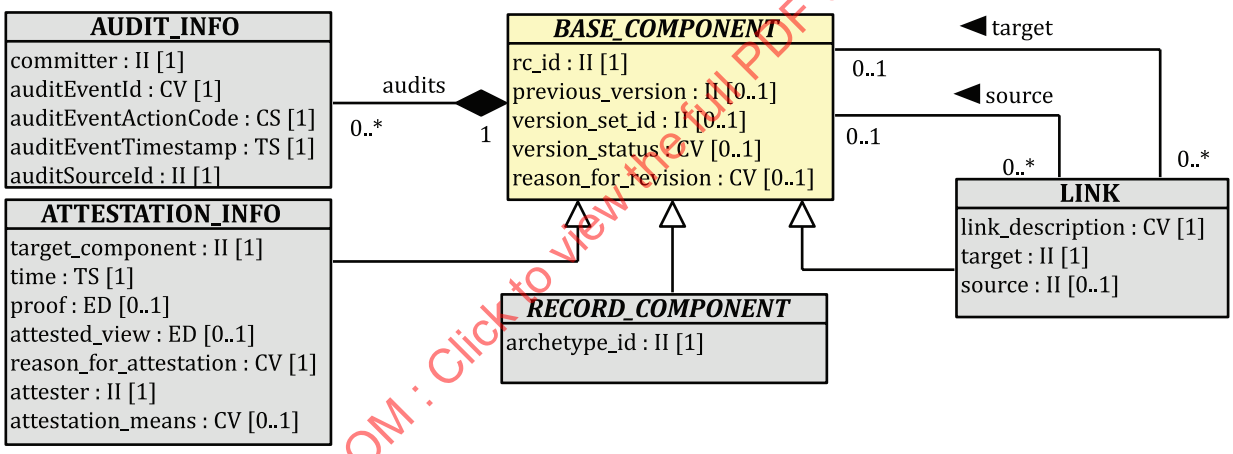

Figure 2 — UML diagram showing all direct relationships of the class BASE_COMPONENT

Attributes

#	Attribute name	Datatype	Description
1	rc_id	II[1]	The globally-unique identifier by which this instance of a BASE_COMPONENT is referenced in the EHR system to which the data were first committed. This identifier shall be retained by the EHR Recipient and re-used whenever this BASE_COMPONENT is subsequently included in another extract.
2	previous_version	II [0..1]	This attribute is the rc_id of the BASE_COMPONENT of which the current BASE_COMPONENT is a revision. If this attribute is null, there is no previous version (i.e. it is the very first version).
3	version_set_id	II [0..1]	This attribute value is the rc_id of the very first version of this BASE_COMPONENT. This attribute may be null if this BASE_COMPONENT is the very first version.

#	Attribute name	Datatype	Description
4	version_status	CV[0..1]	The medico-legal status of this version of the RECORD_COMPONENT. A code set for this attribute is defined in Part 3 of this series.
5	reason_for_revision	CV[0..1]	A code for the reason for assigning the current version status.

Relationships

#	Generalization of
1	RECORD_COMPONENT
2	LINK
3	ATTESTATION_INFO

#	Class name		Name of relationship	Aggregation of	
4	1	BASE_COMPONENT	audits	0..*	AUDIT_INFO

#	Association from		Name of relationship	Association to	
5	0..*	LINK	target	0..1	BASE_COMPONENT
6	0..*	LINK	source	0..1	BASE_COMPONENT

6.1.3 Electronic health record component

Term: *electronic health record component*

Admitted terms: record component, electronic record component

Definition: *health record component* which only includes *information* in electronic format

[Source: ISO 13940]

Class name: RECORD_COMPONENT

Description: This is an abstract class that include the properties that is common for all kinds of record components.

A model showing the associations of this class is shown in Figure 3.

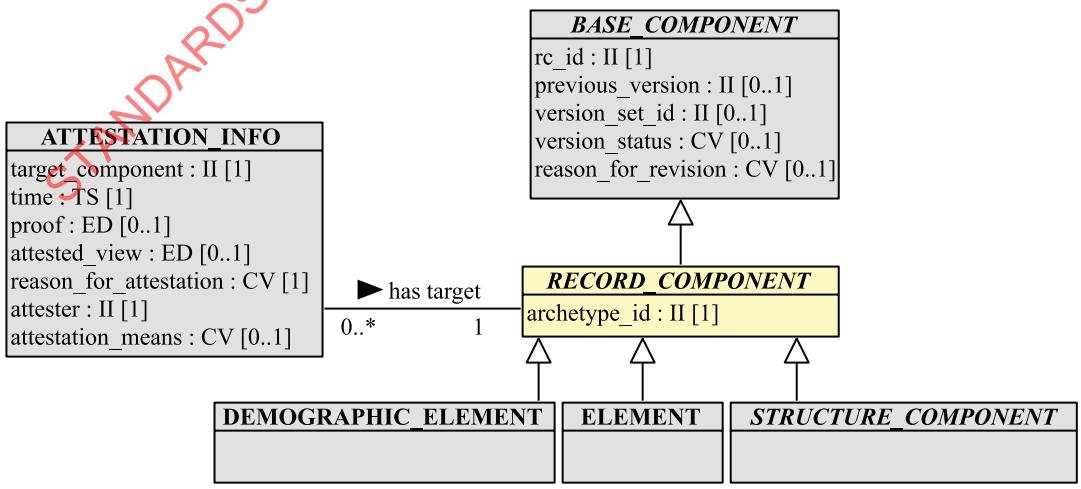

Figure 3 — UML diagram showing all direct relationships of the class RECORD_COMPONENT

Attributes

#	Attribute name	Datatype	Description
1	archetype_id	II[1]	The unique identifier of the archetype node to which this RECORD_COMPONENT corresponds, either in the EHR Provider system or as a mapping produced when this extract was created. The syntax for populating this attribute value is defined in Part 2 of this series.

Relationships

#	Specialization of
1	BASE_COMPONENT

#	Generalization of
2	STRUCTURE_COMPONENT

#	Association from		Name of relationship	Association to	
3	0..*	ATTESTATION_INFO	has target	1..*	RECORD_COMPONENT

This class inherits all attributes and relationships from the following class.

1) BASE_COMPONENT

6.1.4 Structure component

Term: *structure component*

Definition: *electronic health record component* which contain one of more other *electronic health record components*

Class name: STRUCTURE_COMPONENT

Description: This is an abstract class that include the properties that is common for those kinds of record components that is used for building hierarchical structures.

A model showing the associations of this class is shown in Figure 4.

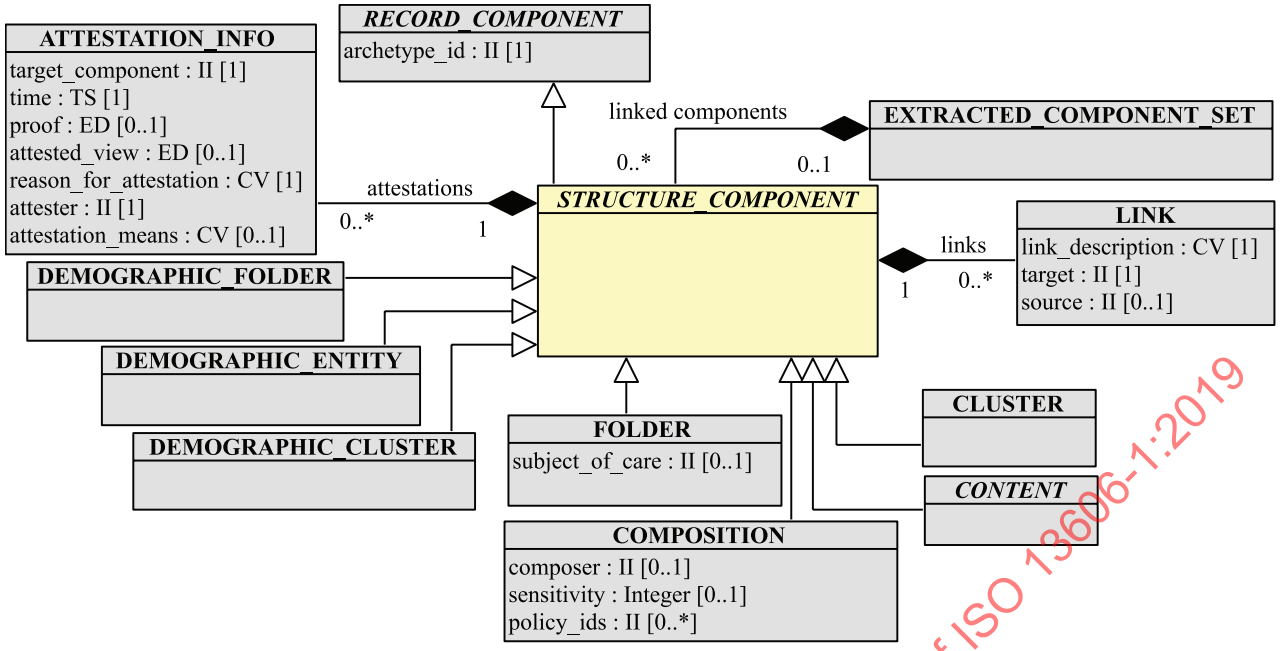

Figure 4 — UML diagram showing all direct relationships of the class STRUCTURE_COMPONENT

Relationships

#	Specialization of
1	RECORD_COMPONENT

#	Generalization of
2	FOLDER
3	COMPOSITION
4	CONTENT
5	CLUSTER
6	DEMOGRAPHIC_FOLDER
7	DEMOGRAPHIC_ENTITY
8	DEMOGRAPHIC_CLUSTER

#	Class name		Name of relationship	Aggregation of	
9	1	STRUCTURE_COMPONENT	attestations	0..*	ATTESTATION_INFO
10	1	STRUCTURE_COMPONENT	links	0..*	LINK
11	0..1	EXTRACTED_COMPONENTS	structure_components	0..*	STRUCTURE_COMPONENT

This class inherits all attributes and relationships from the following classes.

1) RECORD_COMPONENT

2) BASE_COMPONENT (indirect through RECORD_COMPONENT)

6.2 Attestation

6.2.1 General

Attestation is the activity of recording and certifying legal responsibility for a particular unit of information. The attestation of part of an EHR is a mechanism whereby the attester can provide his

or her authority that those contents are, in his or her opinion, correct. This means that he or she is satisfied that the contents are a fair and faithful reflection of the processes they document, and do not deliberately misrepresent the truth. Attesting a part of an EHR will not have modified its content or interpretation, other than by adding weight to its authenticity. (Anything which added an opinion, a new viewpoint or perspective should have been either a revision or a new set of entries with a link to this one.)

Clearly any modification to a part of an EHR through revision cannot automatically carry forward any previous attestations - if necessary the original attester would have been invited to re-attest that he or she remains happy now it has been modified, or the reviser may attested the new version, or both, or neither may attest the new version.

There has been much debate over many years about what information needs to be retained within electronic systems:

a) to verify the authorisation of the attester (ranging from a simple flag to indicate that he had been authenticated in that system's normal way, to a complex hash of the user's digital key, date and time, and part or all of the document being signed, and optionally sent to a trusted third party notary service);

b) as a permanent legal record of what was attested (ranging from no specific addition to the raw database record that is being signed, to XML output files with a style-sheet as a proxy to show how it was presented, to bitmaps of each screen as it was actually presented for signature).

Attestation may be carried out by more than one person, at different times from the committal, and might not always be required in some healthcare activities. The attester will sometimes also be the committer, but might not be, for example, if a medical secretary is typing in the data.

This document acknowledges that in some situations it will be appropriate to communicate the detailed evidence of an attestation, and in others to simply confirm that the data were attested in the EHR Provider's system and to only communicate the name of the signatory and date of the attestation.

The ATTESTATION_INFO class represents the following information about an attestation:

— the date and time at which the attestation occurred;

— an identifier for the person who made this attestation;

— optionally, the means whereby that person carried out the attestation;

— optionally, the list of RECORD_COMPONENTS that were attested, if not simply the content of the class containing this attestation;

— the reason for, or legal significance of, this attestation;

— optionally, the status of this attestation; this may be required in the event of an attestation having been made in error and needing to be withdrawn;

— optionally the electronic signature (as encapsulated data, or as reference to it) that verifies the attestation;

— optionally the encapsulated data, or a reference to it, that represents the visual image that was actually viewed by the attester; it is now required in some EU countries that this is retained within the EHR in addition to the data in its processable form.

Although it may be most common for attestation to be applied to the whole of a COMPOSITION, and to be communicated alongside or as part of the COMPOSITION, this is not universally the case. This document therefore permits attestation information to be contained within any kind of STRUCTURE_COMPONENT, and to list any kinds and number of RECORD_COMPONENTs as having been attested. This might, for example, enable a single composition containing one attestation to effectively authorise a whole set of previously recorded decisions and proposals for care management that are documented in other COMPOSITIONs.

6.2.2 Attestation information

Term: *attestation information*

Definition: *base component* documenting the details of an attestation of a set of *electronic health record components*

Class name: ATTESTATION_INFO

Description: This class documents the details of any attestations that pertain to any RECORD_COMPONENT.

A model showing the associations of this class is shown in Figure 5.

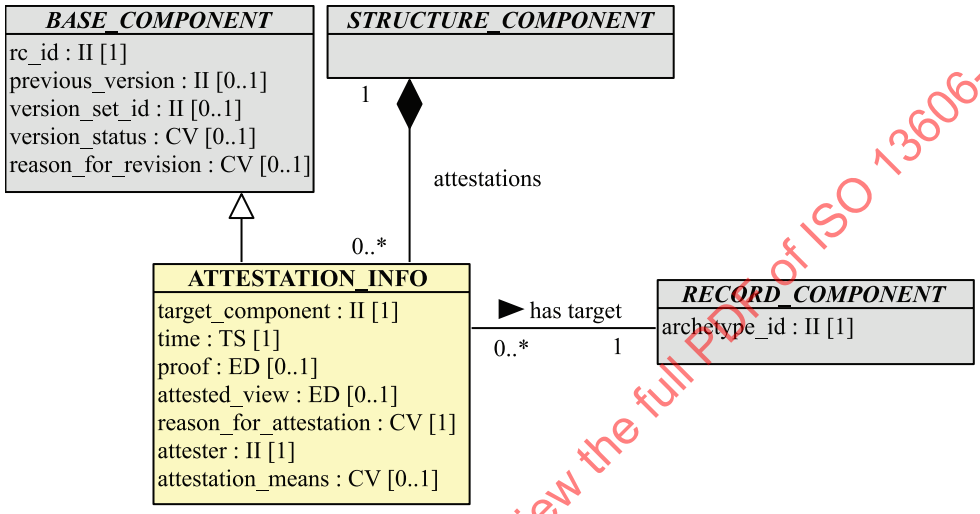

Figure 5 — UML diagram showing all direct relationships of the class ATTESTATION_INFO

Attributes

#	Attribute name	Datatype	Description
1	target_component	II[1]	The record component that is attested. This attribute is shown in the UML diagram as the association 'has target'.
2	time	TS[1]	The date and time at which this attestation occurred.
3	proof	ED [0..1]	The electronic signature (as encapsulated data, or as reference to it) that verifies the attestation. This is optional as it may not always be required when communicating EHR_EXTRACTS, particularly within a single health service.
4	attested_view	ED [0..1]	The encapsulated data, or a reference to it, that represents the reproducible rendering (image or presentation specification) that was actually viewed by the attester.
5	reason_for_attestation	CV[1]	A coded value giving the reason for this attestation, to define its specific purpose or the legal requirement it meets.
6	attester	II[1]	The identification and the structural role of the person making the attestation.
7	attestation_means	CV[0..1]	The mechanism by which the attestation was made.

Relationships

#	Specialization of
1	BASE_COMPONENT

#	Class name		Name of relationship	Aggregation of	
2	1	STRUCTURE_COMPONENT	attestations	0..*	ATTESTATION_INFO

#	Association from		Name of relationship	Association to	
3	0..*	ATTESTATION_INFO	has target	1	RECORD_COMPONENT

This class inherits all attributes and relationships from the following class.

1) BASE_COMPONENT

6.3 Audit information

Term: *audit information*

Definition: documentation of the details of a revision of a *base component*

Class name: AUDIT_INFO

Description: This class documents the details of any revisions that pertain to any BASE_COMPONENT.

NOTE The audit information is a profile on ISO 27789:2013 Health informatics — Audit trails for electronic health records. The profile includes only the minimum set of attributes needed in EHR extracts.

A model showing the associations of this class is shown in Figure 6.

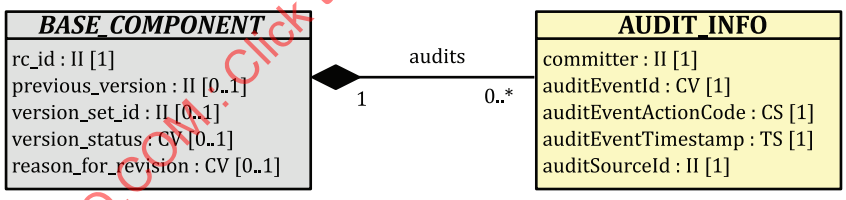

Figure 6 — UML diagram showing all direct relationships of the class AUDIT_INFO

Attributes

#	Attribute name	Datatype	Description
1	committer	II[1]	Unique identifier for the user actively participating in the event. This attribute corresponds to the field UserID in ISO 27789:2013. **Text from ISO 27789:2013:** User identifier text string from the authentication system. It is a unique value within the Audit Source ID. This attribute ties an audit event to a specific user. In this context, a user may be a person, group, team, server, process, or task thread.

#	Attribute name	Datatype	Description
			NOTE 1 For cross-system audits, especially with long retention, this user identifier is meant to permanently tie an audit event to a specific user via a unique key that retains its uniqueness over the entire lifetime of the archiving of the audit trail. NOTE 2 For node-based authentication – where only the system hardware or process, but not a human user, is identified – User ID would be the node name. NOTE 3 If the audit trail is to be used for clinical audit, or to provide evidence, where needed, of misuse, the audit trail might need to record sufficient information to unambiguously associate a unique identifier with an actual user.
2	auditEventId	CV[1]	Code identifying a specific audited event. NOTE This attribute corresponds to the field EventID in ISO 27789:2013. **Text from ISO 27789:2013:** Code identifying a specific audited event, e.g. a menu item, program, rule, policy, function code, application name or URL. It identifies the performed function. Coded value, either defined by the system implementers or as a reference to a standard vocabulary. The "code" attribute shall be unambiguous and unique, at least within Audit Source ID. This identifies the audited function. For "Execute" Event Action Code audit records, this identifies the application function performed. At least one of CodeSystem (OID) or CodeSystemName is mandatory. EXAMPLES Program name, method name or function name.
			NOTE The coding is modelled after IHE ITI TF-1 and TF-2[12] and ISO 12052,[1] DICOM supplement 95. For implementation-defined coded values or references to standards, the XML schema in RFC 3881 defines the optional attributes as shown in Table 4. To support the requirement for unambiguous event identification, multiple values may not be specified. This identifies the audited function. For "Execute" Event Action Code audit records, this identifies the application function performed. At least one of CodeSystem (OID) or CodeSystemName is mandatory.

#	Attribute name	Datatype	Description
3	auditEventActionCode	CS[1]	Indicator for type of action performed in the audit event. Only audit events where the action is create, update or delete shall be included in the EHR extract. NOTE This attribute corresponds to the field EventActionCode in ISO 27789:2013. **Text from ISO 27789:2013:** Description: Indicator for type of action performed in the audit event. — C Create - Create a new database object, such as Placing an Order — R Read/View/Print/Query - Display or print data, such as a diagnosis — U Update - Update data, such as Revise Personal Health Information — D Delete - Make items unaccessible — E Execute - Perform a system or application function such as search, extract, or use of an object's method This broadly indicates what kind of action was done on the Participant Object. NOTE 1 Actions that are not enumerated above are considered an Execute of a specific function or object interface method or treated two or more distinct events. An application action, such as an authorization or digital signing, is a function Execute, and the Event ID would identify the function. NOTE 2 For some applications, such as radiological imaging, a Query action can only determine the presence of data, but not access the data themselves. Auditing need not always make as fine a distinction. NOTE 3 Compound actions, such as "Move," "Archive" or "Copy", would be audited by creating audit data for each operation - read, create, delete - or as an Execute of a function or method.
4	auditEventTimestamp	TS[1]	Date/time of the audited event occurrence. NOTE This attribute corresponds to the field EventDateTime in ISO 27789:2013. **Text from ISO 27789:2013:** A date/time specification that is unambiguous as to local time zones. A date/time representation that is unambiguous in conveying universal coordinated time (UTC). The time shall be in a UTC format, as in ISO 8601:2004 and shall be within a tolerance of no more than 250 ms of UTC. This ties an event to a specific date and time. Security audits typically require a consistent time base to eliminate time-zone issues arising from geographical distribution. NOTE In a distributed system, some sort of common time base, e.g. an NTP [RFC1305] server, is a good implementation tactic.

#	Attribute name	Datatype	Description
5	auditSourceId	II[1]	Identifier of the EHR system where the event originated. This field ties the event to a specific source system. It may be used to group events for analysis according to where the event occurred. NOTE This attribute corresponds to the field AuditSourceID in ISO 27789:2013.

Relationships

#	Class name		Name of relationship	Aggregation of	
1	1	BASE_COMPONENT	audits	0..*	AUDIT_INFO

6.4 Linking record components

6.4.1 General

A user may wish to create ad hoc semantic links between any arbitrary points in an EHR, for example to indicate the evolution of a condition, the likely historic cause of a problem or a response to a previous request, to indicate cause and effect, to track the evolution of orders from request to completion, to monitor outcomes, or to form linkage networks for clinical problems or episodes. In these situations a mechanism is required for a composer to point from any node in their current screen form or electronic document to any previous component in the EHR, and to label the link with an appropriate clinical term. Sometimes one location in the EHR may act as a kind of linkage hub, for example the formal statement of a clinical condition might be used as an anchor point for all historic and subsequent entries relating to that condition (e.g. in a problem oriented record).

Such links will often be created by the user as a pointer from the current record entry to a pre-existing one (the target), or might be created as a new statement of a relationship between two pre-existing entries (by pointing to the source and the target). A wide range of end user interfaces can be envisaged for such functionality, but the task of this document is to provide a generic and safe means for communicating the existence of such links to diverse EHR systems.

A STRUCTURE_COMPONENT may contain any number of instances of LINK, each of which can reference one target component and optionally one source component.

An attribute is available to label each LINK. An informative list of terms for this is given in Part 3 of this series, but other terminologies may be used.

The LINK class only establishes links to other BASE_COMPONENT's in this EHR_EXTRACT. These linkage relationships will most commonly be established between RECORD_COMPONENTs (i.e. between clinical statements), but they may also link to and between attestations (for example, if two or more persons have separately endorsed a significant care decision), and also link to and between other LINKs.

A specialisation of LINK, the EXTERNAL_LINK class, can be used to reference data EHR that is external to this EHR_EXTRACT. This might be used, for example, to reference data within the record of this subject of care but which is not part of the EHR_EXTRACT being communicated, or exists within the record another subject of care (subject to any governing access control rules).

6.4.2 Link usages

6.4.2.1 Referencing a record component in the same EHR

It shall be possible to create a link from a structure component to a record component belonging to the same EHR, by including a LINK instance in the source STRUCTURE_COMPONENT instance. Examples of this kind of health related link might be to associate a treatment with a clinical indication, a complication or side effect with a previous prescription, to connect a series of health situations with an underlying

condition or to establish a health thread. The attribute LINK.target shall contain the rc_id of the target RECORD_COMPONENT instance.

NOTE 1 Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

NOTE 2 This method cannot be used order to create a link *from* an ELEMENT instance since elements deliberately cannot contain links. The LINK instance has to be included in the STRUCTURE_COMPONENT instance which includes the ELEMENT instance. This is described in 6.4.2.2.

6.4.2.2 Creating a link between two record components in the same EHR

It shall be possible to create a link between two record components belonging to the same EHR, by including a LINK instance in a STRUCTURE_COMPONENT instance that is neither the source nor the target of the link. The attribute LINK.source shall contain the rc_id of the source RECORD_COMPONENT instance and the attribute LINK.target shall contain the rc_id of the target RECORD_COMPONENT instance.

NOTE Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

6.4.2.3 Re-using an already existing record component

It shall be possible to include an already existing EHR record component in a new structure component in the same EHR without having to create a copy of the record component that is to be re-used. When doing so, a LINK instance shall be included in the STRUCTURE_COMPONENT instance, and the attribute LINK.target shall contain the rc_id of the RECORD_COMPONENT instance that is to be re-used. Furthermore, the attribute LINK.role shall have the value "included by reference".

6.4.2.4 Referencing a record component in another EHR concerning the same subject of care

It should be possible to create a link from a structure component in an EHR to a record component in another EHR concerning the same subject of care, by including an EXTERNAL_LINK instance in the source STRUCTURE_COMPONENT instance. When creating such a link, the attributes of the EXTERNAL_LINK instance shall be used as follows:

— **target_system** shall contain the identity of the EHR system which includes the target EHR;

— **target_information_type** shall contain a code to the record component type (specialisation) of the target, taken from the set of codes defined in Part 3 of this standard series for this purpose;

— **target** shall contain the rc_id of the target RECORD_COMPONENT instance;

NOTE Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

6.4.2.5 Referencing a demographic record component

It shall be possible to create a link from a structure component to a demographic record component within the EHR system or in a separate system, by including an EXTERNAL_LINK instance in the source STRUCTURE_COMPONENT instance. This is the mechanism by which various human, computational and organisational actors can be connected to record components, for example as information sources or as decision makers. When creating such a link, the attributes of the EXTERNAL_LINK instance shall be used as follows:

— **target_system** shall contain the identity of the system which includes the target demographic record component;

— **target_information_type** shall contain a code to the record component type (specialisation) of the target, taken from the set of codes defined in Part 3 of this standard series for this purpose;

— **target** shall contain the rc_id of the target RECORD_COMPONENT instance;

NOTE Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

6.4.2.6 Referencing links

It should be possible to create a link from a structure component to an instance of LINK within the same EHR, by including a LINK instance in the source STRUCTURE_COMPONENT instance. The attribute LINK.target shall contain the rc_id of the target LINK instance.

It should be possible to create a link between two instances of LINK belonging to the same EHR, by including a LINK instance in a STRUCTURE_COMPONENT instance. The attribute LINK.source shall then contain the rc_id of the source LINK instance and the attribute LINK.target shall contain the rc_id of the target LINK instance.

NOTE Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

6.4.2.7 Referencing attestations

It should be possible to create a link from a structure component to an instance of ATTESTATION_INFO within the same EHR, by including a LINK instance in the source STRUCTURE_COMPONENT instance. The attribute LINK.target shall contain the rc_id of the target ATTESTATION_INFO instance.

It should be possible to create a link between two instances of ATTESTATION_INFO belonging to the same EHR, by including a LINK instance in a STRUCTURE_COMPONENT instance. The attribute LINK.source shall then contain the rc_id of the source ATTESTATION_INFO instance and the attribute LINK.target shall contain the rc_id of the target ATTESTATION_INFO instance.

NOTE Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

6.4.2.8 *Referencing non-standardised information in external systems*

It should be possible to create a link from a structure component to an instance of non-standardised information in an external system, by including a EXTERNAL_LINK instance in the source STRUCTURE_COMPONENT instance. When creating such a link, the attributes of the EXTERNAL_LINK instance shall be used as follows:

— **target_system** shall contain the identity of the system which includes the target information;

— **target** shall contain an identifier that uniquely references the target information within that system;

— **target_information_type** shall contain a code that describes the type of information of the target, taken from the set of codes defined in Part 3 of this standard series for this purpose;

NOTE Part 3 of this standard series includes a set of codes than can be used to describe the role of the link.

6.4.3 Communicating referenced record components

When an instance of EHR_EXTRACT is created, all instances of LINK that are a part of that extract and that are referencing record components that aren't included in the extract, shall be automatically converted to instances of EXTERNAL_LINK.

When an instance of EHR_EXTRACT is created, it should be possible to include a referenced STRUCTURE_COMPONENT instance in an EHR extract, even if that instance isn't a part of a COMPOSITION instance in that extract. The class EXTRACTED_COMPONENT should be used for this purpose: to include the additional referenced components such that all of the references to them can be resolved by the EHR recipient system.

6.4.4 Link

Term: *link*

Definition: *base component* that describes an association from information in an *EHR extract* to other information within that *EHR extract*

Class name: LINK

Description: The LINK class defines the semantics of a non-containment relationship between two BASE_COMPONENTs.

A model showing the associations of this class is shown in Figure 7.

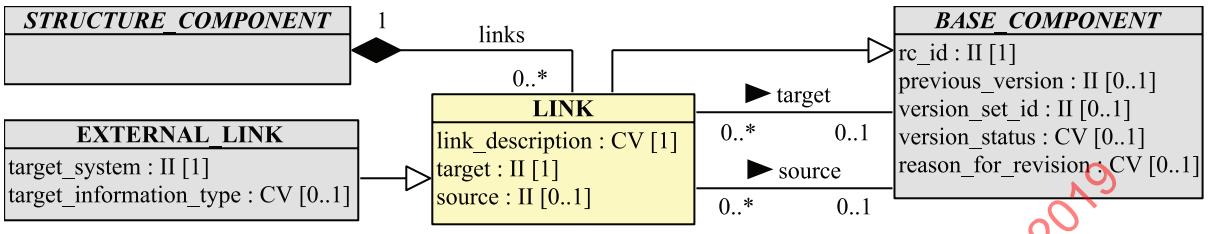

Figure 7 — UML diagram showing all direct relationships of the class LINK

Attributes

#	Attribute name	Datatype	Description
1	link_description	CV[1]	The detailed semantic description of the relationship of the target RECORD_COMPONENT to the source RECORD_COMPONENT. An optional code set for this attribute is defined in Part 3 of this series.
2	target	II[1]	rc_id to the target component. Note: This attribute is shown in the UML diagram as the association 'target'.
3	source	II [0..1]	rc_id to the source component. NOTE This attribute is shown in the UML diagram as the association 'source'.

Relationships

#	Specialization of
1	BASE_COMPONENT

#	Generalization of
2	EXTERNAL_LINK

#	Classname	Name of relationship	Aggregation of
3	1 STRUCTURE_COMPONENT	links	0..* LINK

#	Association from		Name of relationship	Association to	
4	0..*	LINK	target	0..1	BASE_COMPONENT
5	0..*	LINK	source	0..1	BASE_COMPONENT

This class inherits all attributes and relationships from the following class.

1) BASE_COMPONENT

6.4.5 External link

Term: *external link*

Definition: *base component* that describes an association from information in an *EHR extract* to information outside that *EHR extract*

Class name: EXTERNAL_LINK

Description: The EXTERNAL_LINK class is a specialisation of LINK that is to be used when crating a non-containment relationship between a BASE_COMPONENT in an EHR extract and information outside that EHR extract.

Functional requirements

1) The EHR system shall provide the possibility to create a relationship from a BASE_COMPONENT in an EHR to demographic information recorded outside that EHR.

2) The EHR system should provide the possibility to create a relationship from a BASE_COMPONENT in an EHR to other kinds of information than demographics recorded outside that EHR.

A model showing the associations of this class is shown in Figure 8.

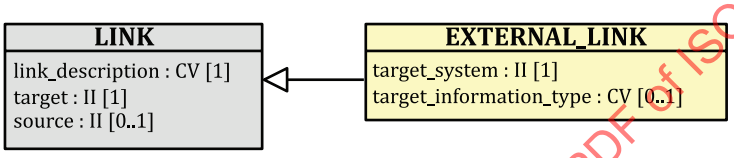

Figure 8 — UML diagram showing all direct relationships of the class EXTERNAL_LINK

Attributes

#	Attribute name	Datatype	Description
1	target_system	II[1]	The identity of the IT system which include the information that is the target of the link.
2	target_information_type	CV[1]	The type of information that is the target of the link. The code set for this attribute is defined in Part 3 of this series. NOTE If the target system is conforming to this document, no value need to be provided.

Relationships

#	Specialization of
1	LINK

This class inherits all attributes and relationships from the following class.

1) LINK

7 Elements and Data Values

7.1 General

This document includes classes for two kinds of elements, ELEMENT and DEMOGRAPHIC_ELEMENT, to represent the leaf nodes within the EHR hierarchy and demographic hierarchy respectively. Each instance of this class will usually have a (single) DATA_VALUE. (A ratio, an interval or a co-ordinated term are considered here to be examples of single data values.) Examples of ELEMENT are reason for encounter, body weight, pulse, haemoglobin level, the dose of a prescribed drug. Examples of DEMOGRAPHIC_ELEMENT are surname, date of birth, post code, telephone number. Both of these specialisations are described later in this document.

This document provides DATA_VALUE data types for:

— text and coded terms;

— quantities including durations;

— dates and time;

— other types of data).

These data types are described below.

If an instance of an ELEMENT is required, for example as a mandatory part of a clinical model, but if the value is not present, the DATA_VALUE may be omitted and a reason for its omission provided using the null_flavor property, taken from the corresponding set of codes defined in Part 3 of this standard series.

7.2 Data value

Term: *data value*

Definition: data contained in an *element*

Class name: DATA_VALUE

Description: Each ELEMENT and DEMOGRAPHIC_ELEMENT contains one data value, to represent the actual instance value being recorded. DATA_VALUE is an abstract class representing the value of an ELEMENT or a DEMOGRAPHIC_ELEMENT. It is instantiated as one of its specialisations, each of which allows a value of a particular datatype to be recorded.

The specializations of DATA_VALUE are the datatypes that shall be used when creating archetypes.

A model showing the associations of this class is shown in Figure 9.

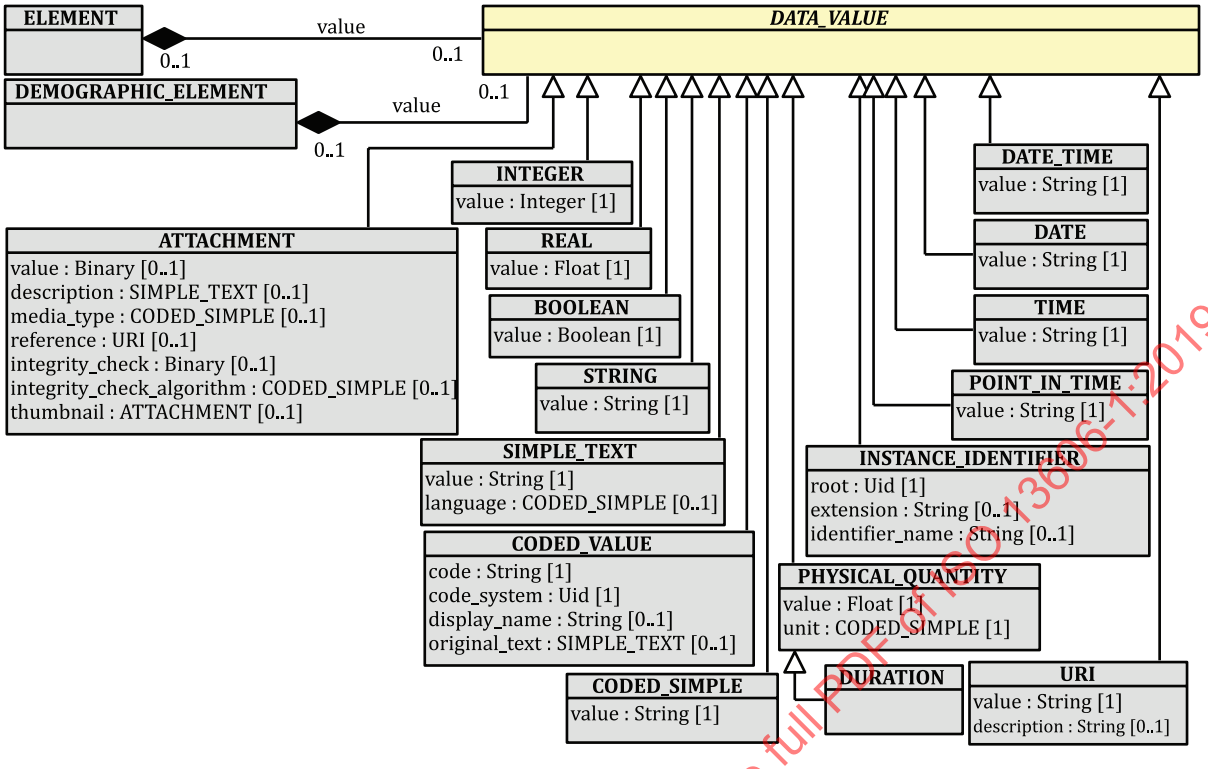

Figure 9 — UML diagram showing all direct relationships of the class DATA_VALUE

Relationships

#	Generalization of
1	ATTACHMENT
2	INTEGER
3	REAL
4	BOOLEAN
5	STRING
6	SIMPLE_TEXT
7	CODED_VALUE
8	CODED_SIMPLE
9	PHYSICAL_QUANTITY
10	DURATION
11	DATE_TIME
12	DATE
13	TIME
14	POINT_IN_TIME
15	INSTANCE_IDENTIFIER
16	URI

#	Class name		Name of relationship	Aggregation of	
1	0..1	ELEMENT	value	0..1	DATA_VALUE

#	Class name		Name of relationship	Aggregation of	
2	0..1	DEMOGRAPHIC_ ELE-MENT	value	0..1	DATA_VALUE

7.2.1 Boolean

Term: *boolean value*

Definition: *data value* used for representing the values of two-valued logic

Class name: BOOLEAN

Description: This DATA_VALUE specialization stands for the values of two-valued logic. A BOOLEAN value can be either true or false.

A model showing the associations of this class is shown in Figure 10.

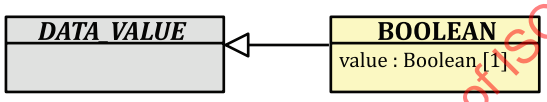

Figure 10 — UML diagram showing all direct relationships of the class BOOLEAN

Attributes

#	Attribute name	Datatype	Description
1	value	Boolean[1]	The value of the BOOLEAN

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.2 Attachment

Term: *attachment value*

Definition: *data value* used for representing information objects that may require special processing operations to be performed in order for the data to be interpreted or rendered

Class name: ATTACHMENT

Description: This DATA_VALUE specialization represents data that is primarily intended for human interpretation or for further machine processing outside the scope of this specification. This includes unformatted or formatted written language, multimedia data, or structured information as defined by a different standard (e.g., XML-signatures).

An attachment can be represented in two forms, inline or by reference. The content is the same whether it is located in-line or remote. Inline data is communicated or moved as part of the attachment value as binary data (a sequence of bytes). By reference data may reside at a different location: a URL/URI that provides reference to the information required to locate the data. Content may be provided in-line and

a by-reference also may be given; in these cases, it is expected that the content of the reference will be exactly the same as the in-line content.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 ED data type. This document only supports the Binary data value (ED.data), whose attribute is renamed to ATTACHMENT.value. The following attributes from the ISO 21090:2011 ED data type are not included in this document:

— ED.value;

— ED.xml;

— ED.compression (This information should be subsumed in the media_type attribute.);

— ED.charset;

— ED.language;

— ED.translation.

In the definition of the attribute ED.description it is stated that "This attribute is not intended to be a complete substitute for the original. For complete substitutes, use the 'translation' property". This limitation is removed in this profile, since the translation attribute is also removed.

A model showing the associations of this class is shown in Figure 11.

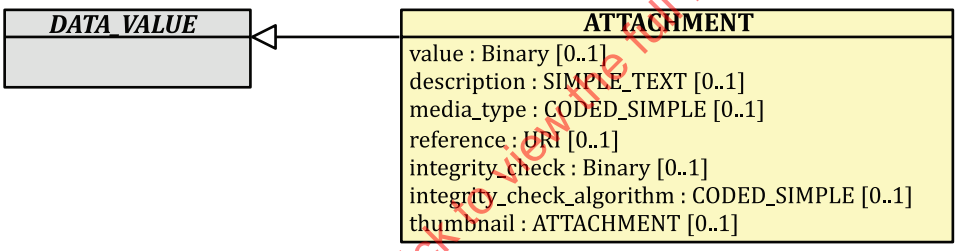

Figure 11 — UML diagram showing all direct relationships of the class ATTACHMENT

Attributes

#	Attribute name	Datatype	Description
1	value	Binary [0..1]	A simple sequence of byte values that contains the content.
2	description	SIMPLE_TEXT [0..1]	An alternative description of the media where the context is not suitable for rendering the media. E.g. Short text description of an image or sound clip. The intent of this property is to allow compliance with disability requirements such as to provide a short text description of included media in some form that can be read by a screen reader.

#	Attribute name	Datatype	Description
3	media_type	CODED_SIMPLE [0..1]	Identifies the type of the attachment and can be used to determine a method to interpret or render the content. The IANA defined domain of media types is established by the IETF RFCs 2045 and 2046. media_type has a default value of text/plain and cannot be null. If the media type is different to text/plain, the media_type attribute shall be populated. If the content is compressed, the media_type shall refer the media_type of the data and to the used compression algorithm, whether the data is accessed by reference or not.
4	reference	URL [0..1]	A reference to a target that provides the binary content. The semantic value of an attachment value is the same, regardless whether the content is present as inline content or just by reference. However, an attachment value without inline content behaves differently, since any attempt to examine the content requires the data to be downloaded from the reference. An encapsulated data value may have both inline content and a reference. If data is provided in the value attribute, the reference shall point to the same data. It is an error if the data resolved through the reference does not match either the integrity check, data as provided, or data that had earlier been retrieved through the reference and then cached. The media_type of the attachment shall match the type returned by accessing the reference.
5	integrity_check	Binary[0..1]	A checksum calculated over the binary data. The purpose of this property, when communicated with a reference is for anyone to validate later whether the reference still resolved to the same content that the reference resolved to when the attachment value with reference was created. If the attribute is null, there is no integrity_check. It is an error if the data resolved through the reference does not match the integrity check. The integrity check is calculated according to the integrity_check_algorithm. By default, the Secure Hash Algorithm-1 (SHA-1) shall be used. The integrity check is binary encoded according to the rules of the integrity check algorithm. The integrity check is calculated over the raw binary data that is contained in the data component, or that is accessible through the reference. No transformations are made before the integrity check is calculated. If the data is compressed, the Integrity Check is calculated over the compressed data.
6	integrity_check_algorithm	CODED_SIMPLE[0..1]	The algorithm used to compute the integrity_check value. If populated, the value of this attribute shall be taken from the HL7 IntegrityCheckAlgorithm code system, OID: 2.16.840.1.113883.5.1010. This is referenced in Part 3 of this standard series.

#	Attribute name	Datatype	Description
7	thumbnail	ATTACHMENT [0..1]	A thumbnail requires significantly fewer resources than the full content, while still maintaining some distinctive similarity with the full content. A thumbnail is typically used with by-reference attachment. It allows a user to select the appropriate content more efficiently before actually downloading through the reference. Originally, the term thumbnail refers to an image in a lower resolution (or smaller size) than another image. However, the thumbnail concept can be metaphorically used for media types other than images. For example, a movie may be represented by a shorter clip; an audio-clip may be represented by another audio-clip that is shorter, has a lower sampling rate, or a lossy compression; or an abstract provided for a long document. A thumbnail itself shall not contain a thumbnail.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.3 String

Term: *string value*

Definition: *data value* used for representing a sequence of Unicode characters

Class name: STRING

Description: This DATA_VALUE specialization stands for text data primarily intended for machine processing (e.g., sorting, querying, indexing, etc.) or direct display. Used for names, symbols, presentation and formal expressions.

A model showing the associations of this class is shown in Figure 12.

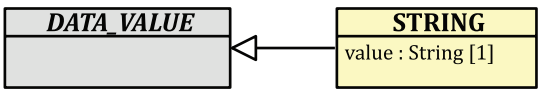

Figure 12 — UML diagram showing all direct relationships of the class STRING

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	The actual content of the string. The String type contains Unicode characters.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.4 Simple text

Term: *simple text value*

Definition: *data value* used for representing a sequence of Unicode characters and optionally the language of the text

Class name: SIMPLE_TEXT

Description: This DATA_VALUE specialization stands for text data primarily intended for machine processing (e.g., sorting, querying, indexing, etc.) or direct display. Used for names, symbols, presentation and formal expressions where there may be a need to include the language of the string.

NOTE This DATA_VALUE specialization is corresponds to a constrained ISO 21090:2011 ST.NT data type where the SN. translation attribute has been removed.

A model showing the associations of this class is shown in Figure 13.

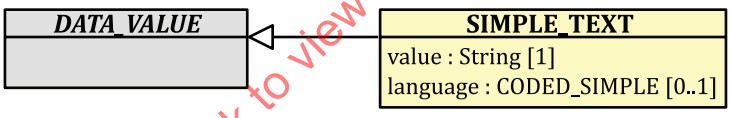

Figure 13 — UML diagram showing all direct relationships of the class SIMPLE_TEXT

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	The actual content of the string. The String type contains Unicode characters.
2	language	CODED_SIMPLE [0..1]	The human language of the content. Valid codes are taken from the IETF RFC 3066[16]. If this attribute is null, the language may be inferred from elsewhere, either from the context or from Unicode language tags, for example.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.5 Coded simple value

Term: *coded simple value*

Definition: *data value* used for representing a code in a code system implied and fixed by the context in which this data value occurs

Class name: CODED_SIMPLE

Description: Coded data in its simplest form, where only the code is not predetermined.

The code system and code system version shall be implied and fixed by the context in which the CS value occurs.

Due to its highly restricted functionality, CODED_SIMPLE shall only be used for simple structural attributes with highly controlled and stable terminologies where:

— All codes come from a single code system;

— Codes are not reused if their concept is deprecated;

— The publication and extensibility properties of the code system are well described and understood.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 CS data type.

A model showing the associations of this class is shown in Figure 14.

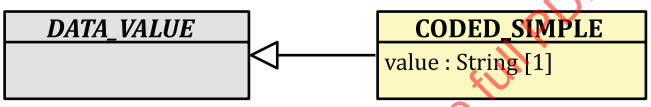

Figure 14 — UML diagram showing all direct relationships of the class CODED_SIMPLE

Attributes

#	Attribute name	Datatype	Description
1	code	String[1]	The plain code symbol defined by the code system. Code shall only contain characters that are either a letter, a digit, or one of '.', '-', '_' or ':'. Code systems that are used with CS shall not define code symbols or expression syntaxes that contain whitespace or any other characters not in this list.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.6 Coded value

Term: *coded value*

Definition: *data value* used for representing a code in a referenced code system

Class name: CODED_VALUE

Description: This DATA_VALUE specialization is a reference to a concept defined in an external code system, terminology, or ontology. A CODED_VALUE may contain a simple code - that is, a reference to a concept defined directly by the referenced code system, or it may contain an expression in some syntax defined by the referenced code system that can be meaningfully evaluated. e.g., the concept of a "left foot" as a post coordinated term built from the primary code "FOOT" and the qualifier "LEFT". CODED_VALUE SHALL be used to represent coded texts.

Together with the code, the code_system shall be provided specifying the system of concepts that defines the code.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 CD.CV data type.

Drafting note: *The property codeSystemName is under consideration for inclusion in this class, and ballot comments are particularly invited on whether this property should be added.*

A model showing the associations of this class is shown in Figure 15.

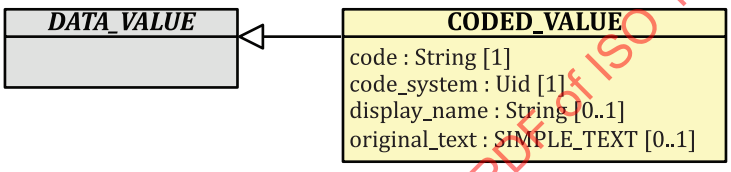

Figure 15 — UML diagram showing all direct relationships of the class CODED_VALUE

Attributes

#	Attribute name	Datatype	Description
1	code	String[1]	The plain code symbol defined by the code system, or an expression in a syntax defined by the code system which describes the concept. If a code is provided, it shall be an exact match to a plain code symbol or expression defined by the code system. If the code system defines a code or expression that includes whitespace, the code shall include the whitespace. An expression can only be used where the code_system either defines an expression syntax, or there is a generally accepted syntax for the code_system. A code system may be defined that only defines an expression syntax with bindings to other code Systems for the elements of the expression. It is at the discretion of the interpreting system whether to check for an expression instead of a simple code and evaluate the expression instead of treating the expression as a code. In some cases, it may be unclear or ambiguous whether the code represents a single symbol or an expression. This usually arises where the code system defines an expression language and then defines pre-coordinated concepts with symbols which match their expression, e.g. UCUM. In other cases, it is safe to treat the expression as a symbol. There is no guarantee that this is always safe: the definitions of the code_system should always be consulted to determine how to handle potential expressions.

#	Attribute name	Datatype	Description
2	code_system	Uid[1]	The code system that defines the code. The code system identification shall include the exact version being used. Code systems shall be referred to by a URI formatted as an UID, which allows unambiguous reference to standard code systems and other local code systems. Where either ISO, CEN, HL7 or other SDOs have assigned UID to code Systems, then these UIDs shall be used. Otherwise implementations SHALL use an appropriate ISO Object Identifier (OID) or UUID to construct a globally unique local coding system identifier.
3	display_name	String [0..1]	A name, title, or representation for the code or expression as it exists in the code system. If populated, the display_name shall be a valid human readable representation of the concept as defined by the code system at the time of data entry. The display_name shall conform to any rules defined by the codingSystem; if the code_system does not define a human representation for the code or expression, then none can be provided. display_name is included both as a courtesy to an unaided human interpreter of a code value and as a documentation of the name used to display the concept to the user. The display name has no functional meaning; it shall never exist without a code; and it shall never modify the meaning of the code. A display name may not be present if the code is an expression for which no display name has been assigned or can be derived. Display names shall not alter the meaning of the code value. Therefore, display names should not be presented to the user on a receiving application system without ascertaining that the display name adequately represents the concept referred to by the code value. Communication shall not simply rely on the display name. The display name's main purpose is to support implementation debugging.
4	original_text	SIMPLE_TEXT [0..1]	The text as seen and/or selected by the user who entered the data which represents the intended meaning of the user. Original text can be used in a structured user interface to capture what the user saw as a representation of the code on the data input screen, or in a situation where the user dictates or directly enters text, it is the text entered or uttered by the user. It is valid to use CODED_VALUE to store only the text that the user entered or uttered. In this situation, original text will exist without a code. In a situation where the code is assigned sometime after the text was entered, original_text is the text or phrase used as the basis for assigning the code. The original text shall be an excerpt of the relevant information in the original sources, rather than a pointer or exact reproduction. Thus the original text shall be represented in plain text form. Values of type CODED_VALUE may have a original text despite not having a code. Any CODED_VALUE with no code signifies a coding exception. In this case, original_text is a name or description of the concept that was not coded.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.7 Instance identifier

Term: *instance identifier value*

Definition: *data value* used for representing identifier that uniquely identifies an object

Class name: INSTANCE_IDENTIFIER

Description: An identifier that uniquely identifies an object.

EXAMPLES EHR identifier, EHR system identifier, order id, service catalogue item id, Vehicle Identification, Number (VIN), driving licence identifier, US social security number, NHS number, Australian medicare number, etc.

NOTE 1 Instance identifiers are defined based on ISO object identifiers.

NOTE 2 An identifier allows someone to select one record, object or thing from a set of candidates. Usually an identifier alone without any context is not usable.

NOTE 3 This DATA_VALUE specialization is equivalent to ISO 21090:2011 II data type. The displayable, scope and reliability attributes have been removed.

A model showing the associations of this class is shown in Figure 16.

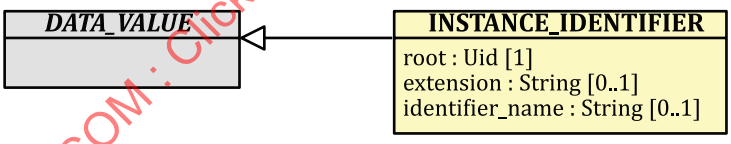

Figure 16 — UML diagram showing all direct relationships of the class INSTANCE_IDENTIFIER

Attributes

#	Attribute name	Datatype	Description
1	root	Uid[1]	A unique identifier that guarantees the global uniqueness of the instance identifier. The value of the root attribute should be either a DCE UUID, an Object Identifier or a special identifier taken from lists published by ISO, CEN, HL7 or other SDOs. NOTE If root is populated, and there is no extension, then the root is a globally unique identifier in its own right. In the presence of a non-null extension, the root is the unique identifier for the "namespace" of the identifier in the extension. Note that this does NOT necessarily correlate with the organization that manages the issuing of the identifiers. A given organization may manage multiple identifier namespaces, and control over a given namespace may transfer from organization to organization over time while the root remains the same.

#	Attribute name	Datatype	Description
2	extension	String [0..1]	A character string as a unique identifier within the scope of the identifier root. NOTE Some identifier schemes define certain style options to their code values. For example, the U.S. Social Security Number (SSN) is normally written with dashes that group the digits into a pattern "123-12-1234". However, the dashes are not meaningful and a SSN can also be represented as "123121234" without the dashes. In the case where identifier schemes provide for multiple representations, the provider organization may make a ruling about which is the preferred form and document that ruling where that respective external identifier scheme is recognized. NOTE Some identifier schemes define certain style options to their code values. For example, the U.S. Social Security Number (SSN) is normally written with dashes that group the digits into a pattern "123-12-1234". However, the dashes are not meaningful and a SSN can also be represented as "123121234" without the dashes. In the case where identifier schemes provide for multiple representations, the provider organization may make a ruling about which is the preferred form and document that ruling where that respective external identifier scheme is recognized.
3	identifier_name	String [0..1]	This is a human-readable name for the namespace represented in the root. NOTE It is a descriptive name for the actual namespace. e.g. "California, U.S. Driver's License Number, 1970-". IdentifierName does NOT refer to the organization which issued the identifier (e.g. California Dept. of Motor Vehicles). It is intended for use as a human readable label when an identifier can be displayed to a human user where an OID would not be meaningful. The identifier name has no computational value. IdentifierName can never modify the meaning of the root attribute. The purpose of the identifer name is to assist an unaided human interpreter of an INSTANCE_IDENTIFIER value to interpret the authority. Applications shall not attempt to perform any decision-making, matching, filtering or other processing based on this presence or value of this property. It is for display and development assistance only. All decision logic shall be based solely on the root and extension properties. Information Processing Entities claiming direct or indirect conformance may choose not to implement identifier_name but shall not reject instances because identifier_name is present. In general, it should only be used when an extension is present, allowing for a display such as "California, U.S. Driver's License Number, 1970-: 123456789". There are absolutely no guidelines for the contents of this text other than it should be completely descriptive of the namespace. E.g. "Driver's License" or even "California Driver's License" would not be ideal. However, formatting, capitalization, whitepace, language, etc. are completely up to the sender.

The concatenation of root and extension values shall be a globally unique identifier for the item that this INSTANCE_IDENTIFIER value identifies.

If no extension attribute is provided in a non-null INSTANCE_IDENTIFIER, then the root shall be the complete unique identifier for the item that this INSTANCE_IDENTIFIER value identifies.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.8 URI

Term: *URI value*

Definition: *data value* used for identifying uniform resource according to the specifications in IETF RFC 2396[15]

Class name: URI

Description: This DATA_VALUE specialization supports the representation of a uniform resource identifier (URI) specified according to IETF RFC 2396[15]. A URI specifies the protocol and the contact point defined by that protocol for the resource.

A URI can be of two types: URLs to identify the location of resources, and URNs to univocally identify a resource.

This specification defines the extensions "ii" to the URN scheme, a reference to an instance identifier (II) value as defined in ISO 21090:2011. The full syntax of the URN is urn:ii:{root}[:{extension}] where {root} and {extension} (if present) are the values from the II that is being referenced.

A model showing the associations of this class is shown in Figure 17.

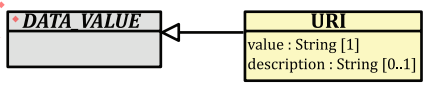

Figure 17 — UML diagram showing all direct relationships of the class URI

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	A URI following the syntax defined in RFC 2396[15]. EXAMPLES — https://www.iso.org/ — urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6 — urn:oid:2.16.724 — urn:ii:2.16.840.1.113883.6.96:73211009
2	description	String [0..1]	Description of the referenced resource in human readable format.

Relationships

#	Specialization of
1	DATA_VALUE

7.2.9 Physical quantity

Term: *physical quantity value*

Definition: *data value* used for representing a quantity expressing the result of a measuring

Class name: PHYSICAL_QUANTITY

Description: This DATA_VALUE specialization represents a dimensioned quantity expressing the result of measuring.

NOTE 1 The general pattern for a measurement is value unit of Thing. In this scheme, the PHYSICAL_QUANTITY represents the value and the unit, and the Thing is described by some coded concept that is linked to the PHYSICAL_QUANTITY by the context of use. This maps obviously to some measurements, such as Subject of care Body Temperature of 37 Celsius, and 250 mg/day of Salicylate.

However, for some measurements that arise in healthcare, the scheme is not so obvious. Two classic examples are 5 Drinks of Beer, and 3 Acetominophen tablets. At first glance it is tempting to classify these measurements like this: 5 drinks of Beer and 3 Acetominophen tablets. The problem with this is that UCUM does not support units of "beer", "tablets" or "scoops".

The reason for this is that neither tablets or scoops are proper units. What kind of tablets? How big is the glass? In these kinds of cases, the concept that appears to be a unit needs to further specify before interoperability is established. If a correct amount is required, then it is generally appropriate to specify an exact measurement with an appropriate UCUM unit. If this is not possible, then the concept is not part of the measurement. UCUM provides a unit called unity for use in these cases. The proper way to understand these measurements as 3 1 Acetominophen tablets, where 1 is the UCUM unit for unity, and the Thing has a qualifier. The context of use will need to provide the extra qualifying information.

NOTE 2 This DATA_VALUE specialization is equivalent to ISO 21090:2011 PQ data type. The translation and coding rationale attributes have been removed.

A model showing the associations of this class is shown in Figure 18.

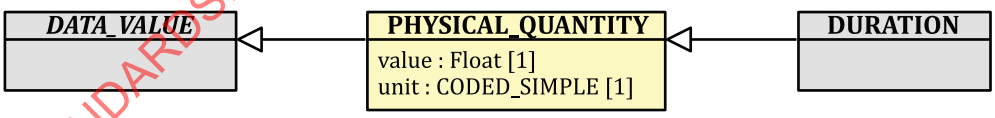

Figure 18 — UML diagram showing all direct relationships of the class PHYSICAL_QUANTITY

Attributes

#	Attribute name	Datatype	Description
1	value	Float[1]	The number which is multiplied by the unit to make the PHYSICAL_QUANTITY value.

#	Attribute name	Datatype	Description
2	unit	CODED_SIMPLE[1]	The unit of measure specified in the Unified Code for Units of Measure (UCUM). UCUM defines two forms of expression, case sensitive and case insensitive. PHYSICAL_QUANTITY uses the case sensitive codes. The code_system OID for the case sensitive form is 2.16.840.1.113883.6.8. The default value for unit is the UCUM code “1” (unity). Equality of physical quantities does not require the values and units to be equal independently. Value and unit is only how we represent physical quantities. For example, 1 m equals 100 cm. Although the units are different and the values are different, the physical quantities are equal. Therefore, one should never expect a particular unit for a physical quantity but instead allow for automated conversion between different comparable units. The unit shall come from UCUM, which only specifies unambiguous measurement units. Sometimes it is not clear how some measurements in healthcare map to UCUM codes.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.10 Duration

Term: *duration value*

Definition: *data value* used for representing a lapse of time

Class name: DURATION

Description: This specialization constrains PHYSICAL_QUANTITY so that it shall have units that describe a period of time

The value of the DURATION is a representation using the minimal unit of measurement. The units shall be a measure of time, according to UCUM ("such as, "s" (second), "min", "h" (hour), "d" (day), "wk" (week), "a" (year)).

NOTE This specialization is equivalent to ISO 21090:2011 PQ.TIME data type. The translation and coding rationale attributes have been removed.

A model showing the associations of this class is shown in Figure 19.

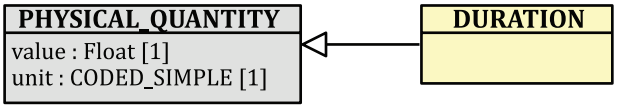

Figure 19 — UML diagram showing all direct relationships of the class DURATION

Relationships

#	Specialization of
1	PHYSICAL_QUANTITY

This class inherits all attributes and relationships from the following class.

1) PHYSICAL_QUANTITY

2) DATA_VALUE

7.2.11 Real

Term: *real value*

Definition: *data value* used for representing a rational number

Class name: REAL

Description: This DATA_VALUE specialization represents a rational number. IT acts as a wrapper of an underlying "primitive" decimal type (real, float).

There are many specifications, languages and implementation technologies that declare similar types to either the underlying real types or the REAL defined in this specification, with a profusion of names around the common theme of Real, Float, Decimal or Double.

This specification does not seek to find new previously unrecognised names for the types it declares that represent these base concepts; instead it just assigns commonly used names to them. Again, this specification is not attempting to redefine or replace any other definitions by using these common names.

For these name clashes between the datatypes defined in this specification and any other specification, implementers should use some form of name spacing to ensure that the names of the datatypes do not cause confusion, perhaps by prefixing the names with some string constant in implementation environments that do not support proper name spacing of types.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 REAL data type.

A model showing the associations of this class is shown in Figure 20.

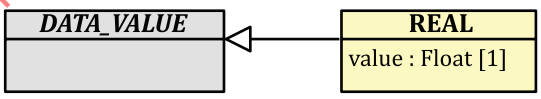

Figure 20 — UML diagram showing all direct relationships of the class REAL

Attributes

#	Attribute name	Datatype	Description
1	value	Float[1]	The value of the REAL represented as a primitive Float of the implementation language.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.12 Integer

Term: *integer value*

Definition: *data value* used for representing an exact integral values.

Class name: INTEGER

Description: This DATA_VALUE specialization is used for precise numbers that are results of counting and enumerating.

Integer numbers are discrete, the set of integers is infinite but countable. No arbitrary limit is imposed on the range of integer numbers. This DATA_VALUE specialization acts as a wrapper of an underlying "primitive" integer type.

EXAMPLES –1, 0, 1, 2, 100, 3398129, etc.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 INTEGER data type.

A model showing the associations of this class is shown in Figure 21.

Figure 21 — UML diagram showing all direct relationships of the class INTEGER

Attributes

#	Attribute name	Datatype	Description
1	value	Integer[1]	The value of the INTEGER represented as a primitive Integer of the implementation language. Note that this specification imposes no limitations on the size of integer, but most implementations will map this to a 32 or 64 bit integer.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.13 Date time

Term: *date time value*

Definition: *data value* used for representing a precise point in time with a resolution of second and optionally fraction of second

Class name: DATE_TIME

Description: This DATA_VALUE specialization represents a quantity specifying a point on the axis of natural time. A point in time is most often represented as a calendar expression.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 TS.DATETIME.FULL data types.

A model showing the associations of this class is shown in Figure 22.

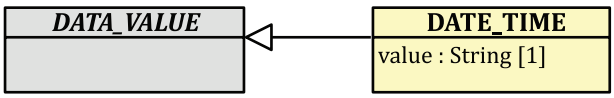

Figure 22 — UML diagram showing all direct relationships of the class DATE_TIME

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	The value of the DATE_TIME is a string following the syntax "YYYY-MM-DDTHH:MM:SS[+|-ZZzz]" defined in ISO 8601.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.14 Date

Term: *date value*

Definition: *data value* used for representing a point in time with a resolution of day of month

Class name: DATE

Description: This DATA_VALUE specialization represents a quantity specifying a point on the axis of natural time that only contains a date value.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 TS.DATE.FULL data type.

A model showing the associations of this class is shown in Figure 23.

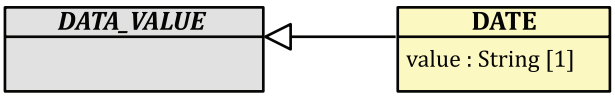

Figure 23 — UML diagram showing all direct relationships of the class DATE

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	The value of the DATE is a string following the syntax "YYYY-MM-DD" defined in ISO 8601.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.15 Time

Term: *time value*

Definition: *data value* used for representing a time of day, without relating it to a particular day

Class name: TIME

Description: This DATA_VALUE specialization represents a quantity specifying a point on the axis of natural time that only contains a time value.

A model showing the associations of this class is shown in Figure 24.

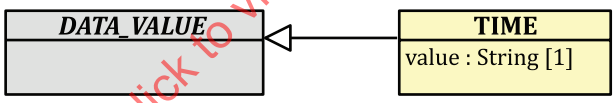

Figure 24 — UML diagram showing all direct relationships of the class TIME

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	The value of the TIME is a string following the syntax "HH:MM:SS" defined in ISO 8601.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

7.2.16 Point in time

Term: *point in time value*

Definition: *data value* used for representing a point in time with minimum resolution year, and maximum resolution fraction of second

Class name: POINT_IN_TIME

Description: This DATA_VALUE specialization represents a quantity specifying a point on the axis of natural time. A point in time is most often represented as a calendar expression.

NOTE This DATA_VALUE specialization is equivalent to ISO 21090:2011 TS.

A model showing the associations of this class is shown in Figure 25.

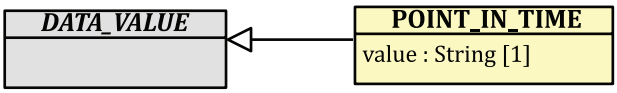

Figure 25 — UML diagram showing all direct relationships of the class POINT_IN_TIME

Attributes

#	Attribute name	Datatype	Description
1	value	String[1]	The value of the POINT_IN_TIME is a string following the syntax "YYYY[-MM[-DD[THH[:MM[:SS[.U[U[U[U]]]]]]]][+|-ZZzz]" defined in ISO 8601. The format should be used to the degree of precision that is appropriate. This attribute can represent partial date time values by removing the less significant portions of the date time value.

Relationships

#	Specialization of
1	DATA_VALUE

This class inherits all attributes and relationships from the following class.

1) DATA_VALUE

8 EHR Components

8.1 General

Figure 26 shows an example version of the reference model for EHR extracts realised as a profile of the reference model described in 5.1. This figure is a subset of the class diagram in Figure 1, showing only those classes that would be used in situations where a shared demographic service is used between the communicating parties, who therefore do not require to communicate detailed demographic information within the EHR_EXTRACT itself. (They will nevertheless need to exchange enough information to allow the recipient to look up the corresponding instances in their own or the shared demographic service.)

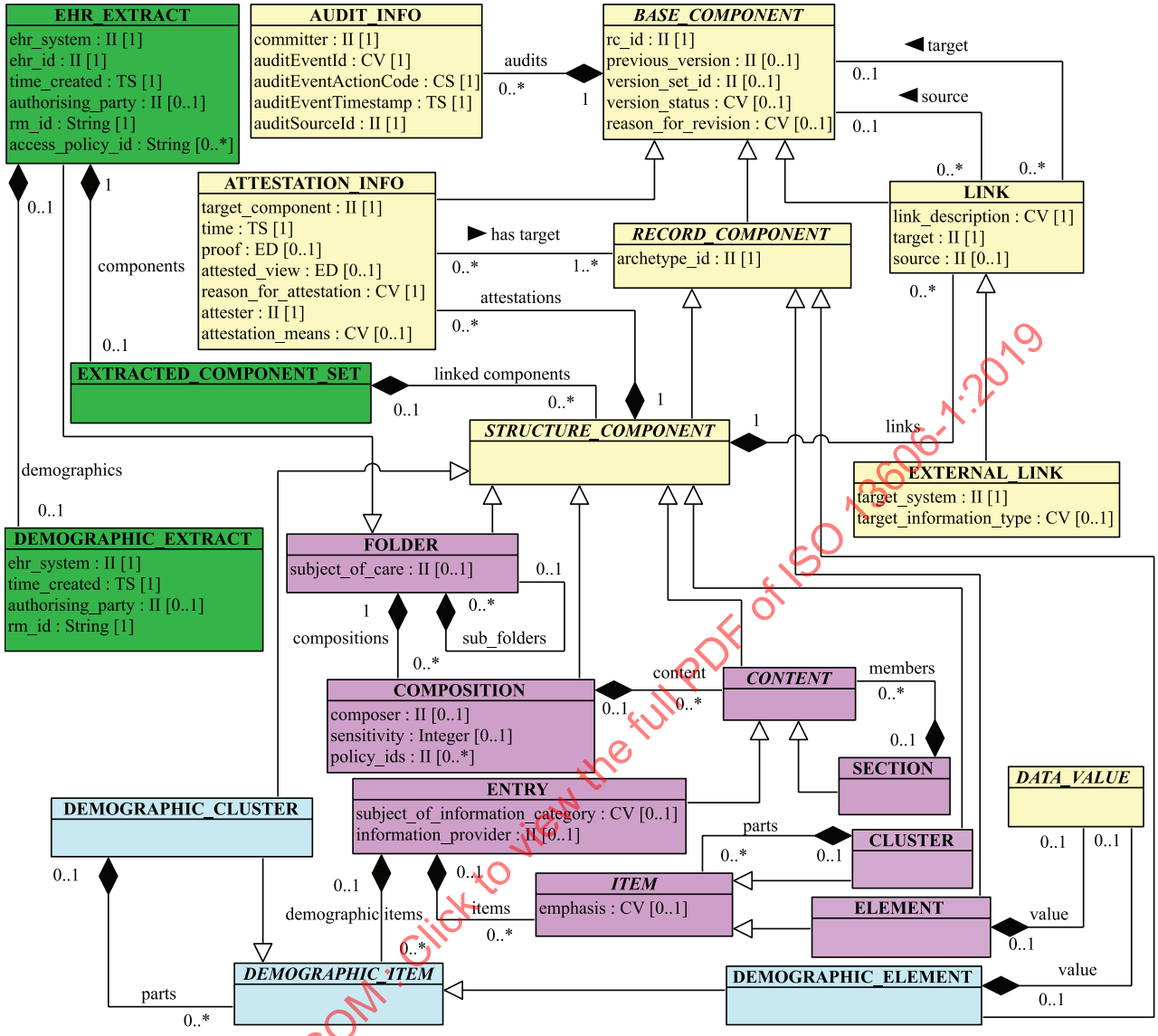

Figure 26 — UML diagram showing a reference model for EHR extract

8.2 Folder

8.2.1 General

This class is used to represent the highest-level organisations of EHR data within the EHR_EXTRACT e.g. to group COMPOSITIONs by episode, care team, clinical speciality, clinical condition or time interval. Internationally, this kind of organising structure is used variably: in some organisations and systems the folder concept is treated as an informal compartmentalisation of the overall health record; in others it might represent a significant legal portion of the EHR relating to the services provided by an enterprise or team.

In the EHR_EXTRACT, FOLDERs are an optional hierarchy. FOLDERs may contain other FOLDERs to form a rich tree structure, and each may include any pertinent information about their committal or revision within the EHR Provider system. FOLDERs can only contain FOLDERs or COMPOSITIONs.

In cases where an EHR_EXTRACT contains the EHR data for more than one subject of care (e.g. for a family), the FOLDER may be used to specify the subject of care of the COMPOSITIONS contained in that FOLDER.

In some situations FOLDERs might be created specifically to organise the EHR_EXTRACT, or contain only a selected subset of the data in the corresponding folder in the EHR provider system. In such circumstances the FOLDERs within the EHR_EXTRACT will not have a direct correspondence with those in the contributing EHR provider system. A special case of this will be the root FOLDER or root FOLDERs within an EHR_EXTRACT, which will specify the subject of care whose EHR data they each contain.

A FOLDER may be used to group a set of COMPOSITIONS comprising the individual records made of members of a multi-professional team during a single clinical encounter. In situations like this where a FOLDER represents a finite interval of care, it may be attested. This approach can be used to communicate that the FOLDER's contents are a complete record of that interval of care. This also provides an indication to the EHR Recipient that additional COMPOSITIONS ought not to be added to this FOLDER.

Since folder systems are used variably within EHR systems, this document cannot prescribe how they should be handled within the EHR Recipient's system: i.e. it does not require that the EHR Recipient explicitly uses these within its EHR system. However, if a FOLDER has been attested an intact copy of this information shall be retained for future reference and possible onward communication.

8.2.2 Folder

Term: *folder*

Definition: *structure component* used for creating hierarchical structures within an *EHR extract* and with *compositions* as leaf nodes

Class name: FOLDER

Description: The FOLDER class may be used to organize and group COMPOSITIONs within an EHR_EXTRACT. This hierarchy might correspond to or resemble the high-level organization of the EHR within the EHR Provider system, or have been created specifically for this EHR_EXTRACT.

A model showing the associations of this class is shown in Figure 27.

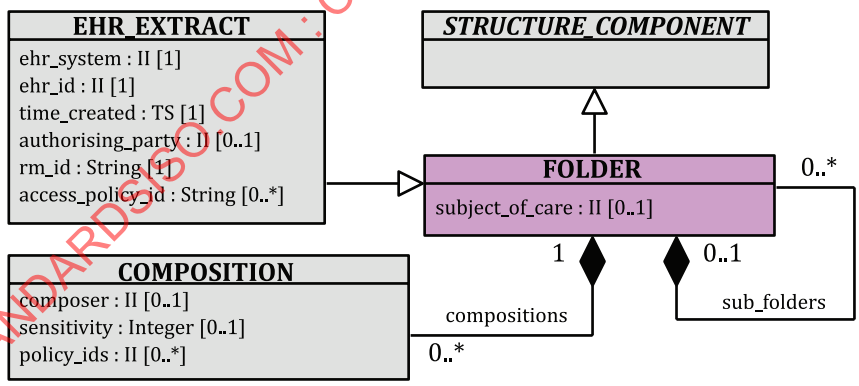

Figure 27 — UML diagram showing all direct relationships of the class FOLDER

Attributes

#	Attribute name	Datatype	Description
1	subject_of_care	II [0..1]	Unique identifier of the subject of care which the healthcare information in that FOLDER instance relates to. NOTE 1 Rules regarding the use of this attribute, are given below. NOTE 2 An instance of FOLDER may include other (sub) FOLDER instances which contain healthcare information relating to other persons than that folder's subject of care. If so, each such (sub) FOLDER instance will have a subject_of_care attribute that identifies the person which the healthcare information in the (sub) FOLDER instance relates to.

For instances of the FOLDER specialization EHR_EXTRACT, the attribute subject_of_care is mandatory and shall always contain a unique identifier of the subject of care from whose EHR this EHR Extract was created, as defined by the EHR Provider system.

If an EHR Extract includes compositions with healthcare information relating to another person than the subject of care from whose EHR this EHR Extract was created, these compositions shall be placed in one or more separate folders. The attribute subject_of_care of such folders, shall always contain a unique identifier of the person which the healthcare information relates to.

Relationships

#	Specialization of
1	STRUCTURE_COMPONENT

#	Generalization of
2	EHR_EXTRACT

#	Class name		Name of relationship	Aggregation of	
3	0..1	FOLDER	sub_folders	0..*	FOLDER
4	1	FOLDER	compositions	0..*	COMPOSITION

This class inherits all attributes and relationships from the following classes.

1) STRUCTURE_COMPONENT

2) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

8.3 Compositions

8.3.1 General

The COMPOSITION represents the set of RECORD_COMPONENTS composed (authored) during one user's clinical session or record documentation session, for committal within one EHR. Common examples of this include a consultation note, a progress note, a report or a letter, an investigation report, a prescription form and a set of bedside nursing observations.

The composer is the agent (party, device or software) responsible for creating, synthesising or organising information that is committed to an EHR. This agent takes responsibility for its inclusion in that EHR, even if not the originator of it and even if not the committer of it. The content of the COMPOSITION is primarily attributed to this person. Whether or not the composer is changed when a revision is made is optional. Applications will generally display the composer's name to attribute the COMPOSITION's data.

There may be occasions when there is no single main composer (e.g. a multi-professional tele-consultation, or a case conference); in such cases the composer role might not be formally specified even though each participant and clinical role is declared. Another situation this document has to cater for is the inclusion of digitally scanned correspondence, for which information about the composer is available for human readership but is not available in a processable form to populate the composer property. The composer property is therefore optional.

Other actors and roles may need to be associated with a COMPOSITION, in addition to the composer. The LINK class may be used to reference an instance of a demographic class that specifies each such additional actor, including devices. Terms to label such links are defined in Part 3 of this standard series.

The COMPOSITION is the main container class for EHR data within the extract itself, to ensure that a consistent containment hierarchy is used within all Extracts. If multiple versions of EHR data are to be communicated within one EHR_EXTRACT, this should be as a set of distinct COMPOSITIONs, each referencing the preceding version and collectively referencing a version set identifier.

8.3.2 Composition

Term: *composition*

Definition: *structure component* containing information composed during one clinical encounter or documentation session, and committed within one *EHR*

Class name: COMPOSITION

Description: A COMPOSITION represents the set of RECORD_COMPONENTS composed (authored) during one clinical encounter or documentation session, and committed within one EHR.

Each instance of COMPOSITION shall include at least one instance of AUDIT_INFO.

A model showing the associations of this class is shown in Figure 28.

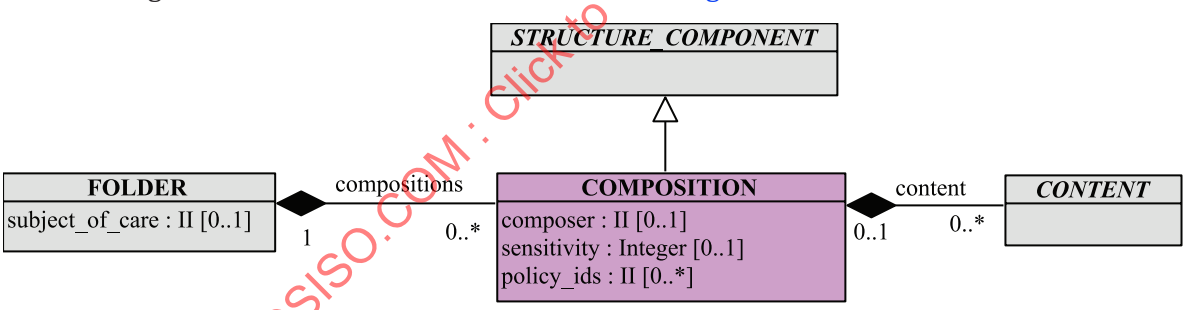

Figure 28 — UML diagram showing all direct relationships of the COMPOSITION

Attributes

#	Attribute name	Datatype	Description
1	composer	II [0..1]	Healthcare actor responsible for creating, synthesising or organising information that is committed to an EHR. The composer takes responsibility for its inclusion in that EHR, even if not the originator of it and even if not the committer of it. The content of the COMPOSITION is primarily attributed to this person. Whether or not the composer is changed when a revision is made is optional, usually depending upon the extent of the changes made.
2	sensitivity	Integer [0..1]	The sensitivity of this COMPOSITION, represented using the code set for this attribute defined in Part 4 of this series.

#	Attribute name	Datatype	Description
3	policy_ids	II [0..*]	This attribute identifies one or more access control policies that specifically pertain to this COMPOSITION and which need to be communicated to the EHR Recipient to govern future access to it. The identifiers may refer to policy information included in this EHR_EXTRACT as defined in Part 4 of this series, or to policies held in external policy servers to which the EHR Recipient has access.

Relationships

#	Specialization of
1	STRUCTURE_COMPONENT

#	Class name		Name of relationship	Aggregation of	
2	1	FOLDER	compositions	0..*	COMPOSITION
3	0..1	COMPOSITION	content	0..*	CONTENT

This class inherits all attributes and relationships from the following classes.

1) STRUCTURE_COMPONENT

2) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

8.4 Content and sections

8.4.1 General

The record entries relating to a single clinical session are usually grouped under headings that represent phases or sub-topics within the clinical encounter or assist with layout and navigation. Clinical headings usually reflect the clinical workflow during a care session and might also reflect the main author's reasoning processes. Research has demonstrated that headings are used differently by different professional groups and specialties, and that headings are not used consistently enough to support safe automatic processing of the EHR. They are therefore treated in this document as an optional (informal) containment for human navigation, filtering and readability.

SECTIONs may be used to represent the containment hierarchy of clinical headings used within the EHR Provider system to group and organise entries within a COMPOSITION. Each SECTION may contain additional SECTIONs and/or ENTRYs.

8.4.2 Content

Term: *content*

Definition: abstract *structure component* that holds all properties common for *sections* and *entries*

Class name: CONTENT

Description: This class is the abstract parent of SECTION and ENTRY, which constitute the EHR data content of a COMPOSITION.

A model showing the associations of this class is shown in Figure 29.

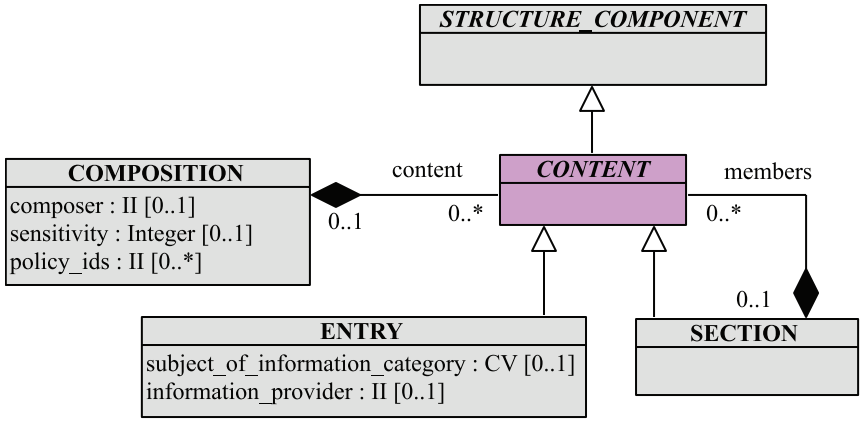

Figure 29 — UML diagram showing all direct relationships of the class CONTENT

Relationships

#	Specialization of
1	STRUCTURE_COMPONENT

#	Generalization of
2	SECTION
3	ENTRY

#	Class name		Name of relationship	Aggregation of	
4	0..1	COMPOSITION	content	0..*	CONTENT
5	0..1	SECTION	members	0..*	CONTENT

This class inherits all attributes and relationships from the following classes.

1) STRUCTURE_COMPONENT

2) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

8.4.3 Section

Term: *section*

Definition: *structure component* used for creating hierarchical structures within a *composition* and with *entries* as leaf nodes

Class name: SECTION

Description: SECTION contains the set of ENTRYs and optionally further SECTIONs that are grouped under one clinical heading.

A model showing the associations of this class is shown in Figure 30.

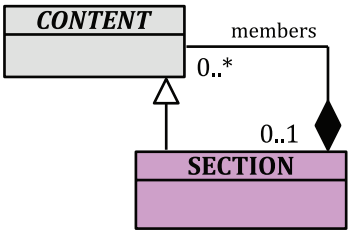

Figure 30 — UML diagram showing all direct relationships of the class SECTION

Relationships

#	Specialization of
1	CONTENT

#	Class name		Name of relationship	Aggregation of	
2	0..1	SECTION	members	0..*	CONTENT

This class inherits all attributes and relationships from the following classes.

1) CONTENT

2) STRUCTURE_COMPONENT (indirect through CONTENT)

3) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

8.5 Entries

8.5.1 General

The ENTRY is the container class for the ITEM data structure that represents the information acquired and recorded for a single observation or observation-set (battery or time series), a single clinical statement such as a portion of the subject of care's history or an inference or assertion, or a single action that might be intended or has actually been performed. The ENTRY class associates this ITEM structure with a set of context attributes to facilitate safe interpretation:

— information in an ENTRY may be about someone other than the subject of care (e.g. a relative): ENTRY defines the subject of the information;

— information in an ENTRY may have been provided by or is attributed to a particular individual: ENTRY defines the information provider;

— other participants might need to be associated with a particular ENTRY.

Multiple actors and roles may therefore need to be associated with an ENTRY. ENTRY may contain instances of DEMOGRPAHIC_ITEM to enable such actors to be represented. The LINK class may also be used to reference instances of a demographic class that specifies such additional actors, including devices. Terms to label such links are defined in Part 3 of this standard series.

The ENTRY contains a data structure represented using CLUSTERs and ELEMENTs. It is important to note that ENTRY cannot contain further ENTRYs. The set of contexts defined at the ENTRY level (e.g. the subject of information) apply to the whole data structure and cannot be over-ridden.

8.5.2 Entry

Term: *entry*

Definition: *structure component* that represents the data documenting one clinical observation or a series of related observations, or a clinical statement, an explanation of clinical reasoning, a healthcare intention, plan or action.

Class name: ENTRY

Description: The ENTRY class contains (as ITEMs) the information acquired and recorded for a single observation or observation-set (battery or time series), a single clinical statement such as a portion of the subject of care's history or an inference or assertion, or a single action that is intended or has actually been performed. An ENTRY may have zero ITEMs if it is a revision of an ENTRY previously recorded in error.

A model showing the associations of this class is shown in Figure 31.

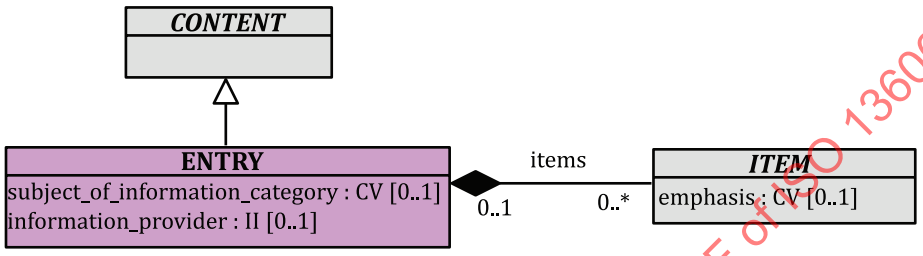

Figure 31 — UML diagram showing all direct relationships of the class ENTRY

Attributes

#	Attribute name	Datatype	Description
1	subject_of_information_category	CV [0..1]	The relationship category of person or object about whom the information in this ENTRY relates to the subject of care. The code set for this attribute is defined in Part 3 of this series. An ENTRY about more than one information subject shall be duplicated per subject within the EHR_EXTRACT.
2	information_provider	II [0..1]	The healthcare provider that provided the information documented in this entry. The information provider does not need to be specified if the information source is implicit from the context.

Relationships

#	Specialization of
1	CONTENT

#	Class name		Name of relationship	Aggregation of	
2	0..1	ENTRY	items	0..*	ITEM

This class inherits all attributes and relationships from the following classes.

1) CONTENT

2) STRUCTURE_COMPONENT (indirect through CONTENT)

3) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

8.6 Items and clusters

The ITEM may represent both the actual data describing the observation, inference, or action, and optionally the details describing the examination method, the subject of care's physical state, or details supporting the clinical reasoning process such as a reference to an electronic guideline, decision support system, or other knowledge reference. Information in an ITEM (CLUSTER or ELEMENT) might have originated at a date/time different from the care activity or its recording, and a specific occurrence date or time may therefore need to be included. The way in which these different sub-components are defined and organised will be specified through archetypes.

Information in an ITEM might be emphasised by the author as being exceptional or noteworthy. This document does not define a code set for this attribute: any agreed terminology may be used to specify the degree of emphasis or to specify the kind of exception.

The CLUSTER supports the representation of complex data structures needed to represent the actual data values within a multi-part (nested) observation, clinical statement, or instruction. These may need to be represented as a table, a tree or a time series. Specific examples include an ECG tracing, a full blood count, ankle reflex examination, the prescription of an intravenous drug infusion.

8.6.1 Item

Term: *item*

Definition: abstract *structure component* that holds all properties common for *clusters* and *elements*

Class name: ITEM

Description: This class is the abstract parent of CLUSTER and ELEMENT, which represent the data structure and values contained by an ENTRY.

A model showing the associations of this class is shown in Figure 32.

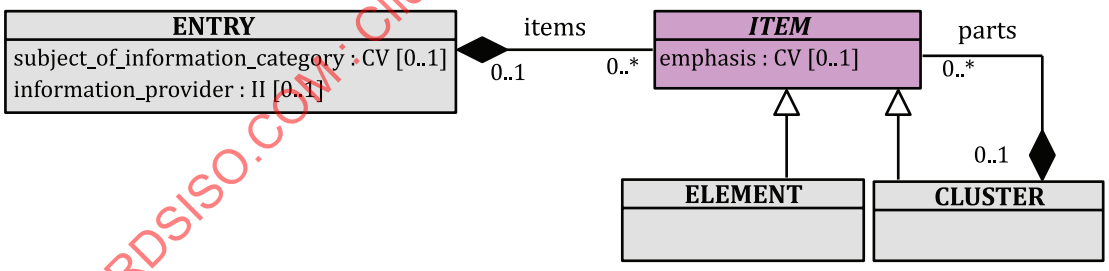

Figure 32 — UML diagram showing all direct relationships of the class ITEM

Attributes

#	Attribute name	Datatype	Description
1	emphasis	CV [0..1]	A way of denoting that the composer wished to mark this ITEM as being of particular note (an unusual measurement value, an unexpected outcome, anything that might be considered necessary to highlight to a future reader).

Relationships

#	Generalization of
1	CLUSTER

#	Generalization of
2	ELEMENT

#	Class name		Name of relationship	Aggregation of	
3	0..1	ENTRY	items	0..*	ITEM
4	0..1	CLUSTER	parts	0..*	ITEM

8.6.2 Cluster

Term: *cluster*

Definition: *structure component* used for creating hierarchical structures within an *entry* and with *elements* as leaf nodes

Class name: CLUSTER

Description: This class represents the hierarchical organisation of the data structure of each ITEM within an ENTRY, to permit the nesting or grouping of ELEMENTs.

A model showing the associations of this class is shown in Figure 33.

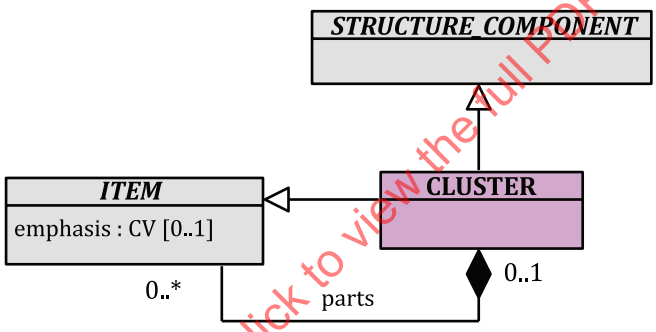

Figure 33 — UML diagram showing all direct relationships of the class CLUSTER

Relationships

#	Specialization of
1	ITEM
2	STRUCTURE_COMPONENT

#	Class name		Name of relationship	Aggregation of	
3	0..1	CLUSTER	parts	0..*	ITEM

This class inherits all attributes and relationships from the following classes.

1) ITEM

2) STRUCTURE_COMPONENT

3) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

8.7 Element

Term: *element*

Definition: *electronic health record component* within an *entry* or *cluster* containing a single *data value*

Class name: ELEMENT

Description: This class represents the leaf nodes within the EHR hierarchy.

A model showing the associations of this class is shown in Figure 34.

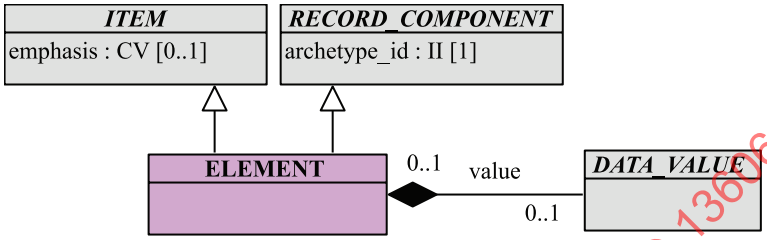

Figure 34 — UML diagram showing all direct relationships of the class ELEMENT

Relationships

#	Specialization of
1	ITEM
2	RECORD_COMPONENT

#	Class name		Name of relationship	Aggregation of	
3	0..1	ELEMENT	value	0..1	DATA_VALUE

This class inherits all attributes and relationships from the following classes.

1) ITEM

2) RECORD_COMPONENT

3) BASE_COMPONENT (indirect through RECORD_COMPONENT)

9 EHR Extract

9.1 General

The EHR_EXTRACT is used to represent part or all of the health record information extracted from an EHR Provider system for the purposes of communication to an EHR Recipient (which might be another EHR system, a healthcare data repository, a client application or a middleware service such as an electronic guideline component), and supporting the faithful inclusion of the communicated data in the receiving system.

The EHR_EXTRACT class contains attributes to identify the EHR Provider organisation and the EHR system from which it has been derived, when it was generated and optionally the agent responsible for creating it.

The EHR_EXTRACT contains the EHR data, in four parts:

1) a directory of FOLDERs that provide a high-level grouping and organising of the COMPOSITIONs. This includes one or more root FOLDERs, each specifying the identity of the subject of care to which that FOLDER's EHR data relate;

2) optionally, a DEMOGRAPHIC_EXTRACT that describes each of the persons or organisations that are referenced from within the RECORD_COMPONENTs contained in this EHR_EXTRACT. This approach allows such entities to be referenced uniquely via an identifier within the body of the EHR, without repetition of the descriptive details each time, and also ensures that any EHR_EXTRACT can be interpreted in isolation if the recipient system does not have access to the services needed to decode the entity identifiers used by the EHR Provider;

3) optionally, a set of EXTRACTED_COMPONENTS that are referenced from, but were not originally composed within, the main RECORD_COMPONENT hierarchy;

4) access policy information, as defined in Part 4 of this series, relating to the information contained within this EHR_EXTRACT. This is intended to inform the EHR Recipient about the wishes of the subject of care and of healthcare providers for how future access requests for the data should be managed.

9.2 EHR extract

Term: *electronic health record extract*

Admitted term: EHR extract

Abbreviation: EHR extract

Definition: *health record extract* consisting solely of *electronic record components*

[Source: ISO 13940:2015]

Class name: EHR_EXTRACT

Description: This class represents the root node of an EHR Extract: part or all of the health record information extracted from an EHR Provider system for the purposes of communication to an EHR Recipient (which might be another repository, a client application or a middleware service such as an electronic guideline component).

A model showing the associations of this class is shown in Figure 35.

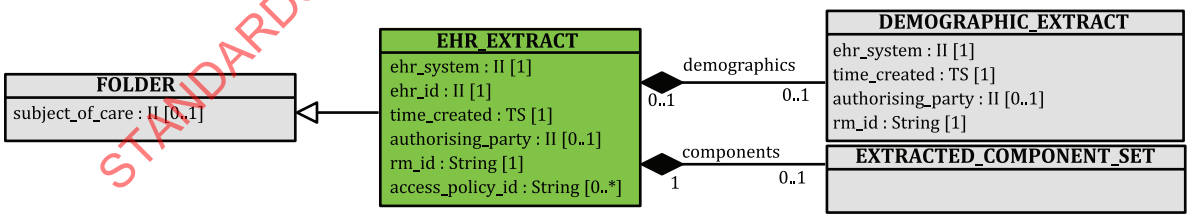

Figure 35 — UML diagram showing all direct relationships of the class EHR_EXTRACT

Attributes

#	Attribute name	Datatype	Description
1	ehr_system	II[1]	The identity of the EHR Provider system from which this EHR Extract has been created. This property identifies the system, and not the provider as a legal entity, as some providers might have more than one system e.g. if they run more than one non-connected site.
2	ehr_id	II[1]	The identity of the EHR from which this EHR Extract has been created. It shall be unique for that EHR Provider system.
3	time_created	TS[1]	The date and time at which data from this subject of care's EHR was queried or exported in order to create this EHR Extract.
4	authorising_party	II [0..1]	Agent authorising the EHR Extract to be created and communicated. This attribute is optional since some extracts might be created automatically between (authorised) interacting computing services.
5	rm_id	string[1]	The identity and version of the Reference Model standard under which this EHR_EXTRACT was made. For an EHR Extract conforming to this part of ISO 13606 the attribute will have the string value "ISO 13606".
6	access_policy_id	string [0..*]	Set of access control policies that pertain to part or all of the data contained in the EHR_EXTRACT. These are included with the EHR_EXTRACT to inform the EHR recipient of the access control measures that ought to apply to these data within the recipient's EHR system and be include with any onward communication of these data by the recipient. The identifiers may refer to policy information included in this EHR_EXTRACT as defined in Part 4 of this series, or to policies held in external policy servers to which the EHR Recipient has access.

Relationships

#	Specialization of
1	FOLDER

#	Class name		Name of relationship	Aggregation of	
2	0..1	EHR_EXTRACT	demographics	0..*	DEMOGRAPHIC_EXTRACT
3	1	EHR_EXTRACT	components	0..1	EXTRACTED_COMPONENTS

This class inherits all attributes and relationships from the following classes.

1) FOLDER

2) STRUCTURE_COMPONENT (indirect through FOLDER)

3) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

9.3 Extracted component set

Term: *extracted component set*

Definition: *structure component* extracted from an EHR for the purpose of being communicated in an EHR extract as the target of one or more links

Class name: EXTRACTED_COMPONENT_SET

Description: An instances of this class is a container for the set of structure components that need to be included in an EHR extract as targets for the links in that EHR extract that references other record components than those otherwise included in that EHR extract.

A model showing the associations of this class is shown in Figure 36.

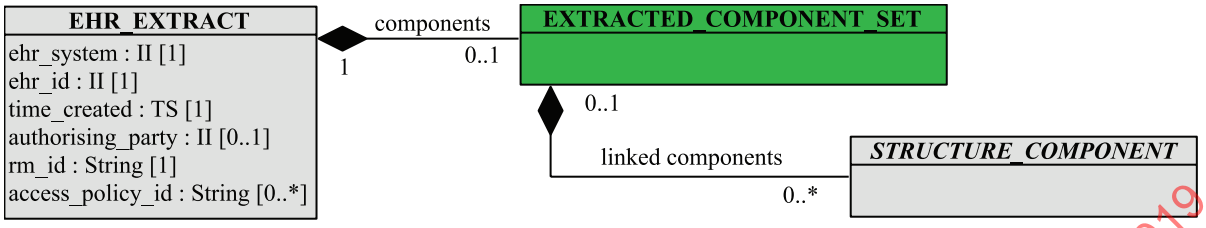

Figure 36 — UML diagram showing all direct relationships of the class EXTRACTED_ COMPONENT_SET

Relationships

#	Class name		Name of relationship	Aggregation of	
1	1	EXTRACTED_COMPONENTS	linked components	0..*	DEMOGRAPHIC_ENTITY
2	0..1	EHR_EXTRACT	components	0..1	EXTRACTED_COMPONENT_SET

9.4 Demographic extract

Term: *demographic extract*

Definition: demographic information extracted for the purpose of being included in an *EHR extract* or an *EHR audit log extract*

Class name: DEMOGRAPHIC_EXTRACT

Description: This class represents the extracted demographics that are to be included in an EHR extract.

A model showing the associations of this class is shown in Figure 37.

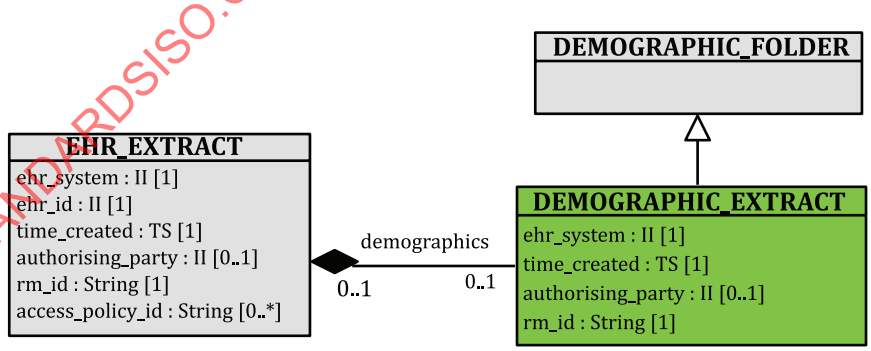

Figure 37 — UML diagram showing all direct relationships of the DEMOGRAPHIC_EXTRACT

Attributes

#	Attribute name	Datatype	Description
1	ehr_system	II[1]	The identity of the EHR Provider system from which this Demographic extract has been created.

#	Attribute name	Datatype	Description
2	time_created	TS[1]	The date and time at which data from this subject of care's EHR was queried or exported in order to create this Demographic extract.
3	authorising_party	II [0..1]	Agent authorising the Demographic extract to be created and communicated. This attribute is optional since some extracts might be created automatically between (authorised) interacting computing services.
4	rm_id	string[1]	The identity and version of the Reference Model standard under which this Demographic extract was made.

Relationships

#	Specialization of
1	DEMOGRAPHIC_FOLDER

#	Class name	Name of relationship	Aggregation of
2	0..1 EHR_EXTRACT	demographics	0..1 DEMOGRAPHIC_EXTRACT

This class inherits all attributes and relationships from the following classes.

1) DEMOGRAPHIC_FOLDER

2) STRUCTURE_COMPONENT (indirect through DEMOGRAPHIC_FOLDER)

3) BASE_COMPONENT (indirect through STRUCTURE_COMPONENT)

10 Demographics

10.1 General

An Electronic Health Record may refer to a wide range of specific entity instances, such as the subject of care, the various healthcare and other agents who have played roles in the delivery of healthcare, devices that have measured body parameters or delivered treatments, and organisations that have assumed responsibilities for care. Many of these entities are referenced many times within any given EHR, and in any enterprise might be defined within a demographics server.

In this Reference Model an equivalent approach has been taken: specific entities are defined once within a demographic extract package, and referenced as necessary throughout the rest of the EHR_EXTRACT by a dedicated instance identifier. The instance identifier used within the EHR_EXTRACT might be, but need not be, one of the actual identifiers by which each entity is known in the EHR Provider system.

The goal of this part of the model is to provide a necessary and sufficient description of each entity to support human interpretation of the EHR, and demographic matching to enable the EHR Recipient to identify the corresponding entities within its own demographic server. Like the EHR data hierarchy, the demographic hierarchy comprises a basic set of building blocks, and requires the use of archetypes to specify and communicate the way in which particular combinations of demographic STRUCTURE_COMPONENTs have been used to organise the descriptors for each kind of entity, such as the way in which addresses are structured. Some commonly occurring demographic archetypes are included in Part 3 of this series.

The whole DEMOGRAPHIC_EXTRACT package is optional, and the demographic details of each entity need not be communicated if it is known that both EHR Provider and EHR Recipient share or can access a common demographic service – for example within one enterprise, region or health service.

Figure 38 shows the reference model for Demographics.