INTERNATIONAL STANDARD

ISO/HL7 21731

First edition
2006-08-01

Corrected version
2012-10-15

Health informatics — HL7 version 3 — Reference information model — Release 1

Informatique de santé — HL7 version 3 — Modèle d'information de référence — Version 1

Reference number
ISO/HL7 21731:2006(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Switzerland
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Health Level Seven International.
Standards Publishing Department
3300 Washtenaw Avenue, Suite 227 • Ann Arbor, MI 48104 • USA
Tel. +1 734 677 77 77
Fax +1 734 677 66 22
E-mail hq@HL7.org
Web www.HL7.org

HEALTH LEVEL SEVEN REFERENCE INFORMATION MODEL, RELEASE 1

Modeling & Methodology Co-Chair	George Beeler, Jr., PhD. Beeler Consulting LLC
Editor	James Case, DVM, PhD. University of California, Davis
Modeling & Methodology Co-Chair	Jane Curry Sierra Systems & CIHI - HL7 Canada
Modeling & Methodology Co-Chair	Ann Hueber Pathology Associates Medical Laboratories
Editor	Lloyd Mckenzie IBM Canada & CIHI - HL7 Canada & Alberta Wellnet
Editor	Gunther Schadow, M.D., PhD. Regenstrief Institute for Health Care
Modeling & Methodology Co-Chair	Abdul-Malik Shakir Shakir Consulting LLC

Health Level Seven, 3300 Washtenaw Avenue, Suite 227, Ann Arbor, MI 48104-4261. Phone: 734-677-7777 Fax: 734-677-6622, Email: HQ@HL7.org; Internet: http://www.HL7.org

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of technical committees is prepare International Standards. Draft International Standards adopted by the technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75 % of the member bodies casting a vote.

A pilot project between ISO and Health Level Seven Inc. (HL7) has been formed to develop and maintain a group of ISO/HL7 standards in the field of medical devices as approved by Council resolution 7/2002. Under this pilot project, HL7 is responsible for the development and maintenance of these standards with participation and input from ISO member bodies.

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.

ISO/HL7 21731 was prepared by HL7 and Technical Committee ISO/TC 215, *Health informatics*.

This corrected version incorporates important new copyright and licensing agreement information on the cover page of this document.

0 Introduction

This introduction is confined to discussion of the requirement for a Reference Information Model in standardization. Further information on the development of this model and the rational for advancing it as a standard can be found in Annex A.

0.1 Uses of a Reference Information Model (RIM) in Health Informatics

0.1.1 Use of the RIM in ISO TC215

ISO TC215 – Health Informatics has previously advanced ISO 17113, a specification for a framweork for developing health data interchange standards. This framework specification calls for messaging standards to be based on a single, comprehensive model of health information. The RIM presented in the current specification provides one such model. Further, the RIM may provide a reference document that can facilitate the harmonization of the health informatics standards and relaterd specifications within ISO TC 215.

0.1.2 Use of the RIM by HL7

The HL7 RIM is a critical component of the V3 development process. It is the root of all information models and structures developed as part of the V3 development process.

The HL7 V3 standard development process is a model-driven methodology in which a network of inter-related models are developed that depict the static and behavioral aspects of the requirements and design of HL7 standards, as well as the underlying semantics and business rules that govern them.

0 INTRODUCTION

The RIM provides a static view of the information needs of HL7 V3 standards. It includes class and state-machine diagrams and is accompanied by use case models, interaction models, data type models, terminology models, and other types of models to provide a complete view of the requirements and design of HL7 standards. The classes, attributes, state-machines, and relationships in the RIM are used to derive domain-specific information models that are then transformed through a series of constraining refinement processes to eventually yield a static model of the information content of an HL7 standard.

The HL7 V3 standard development process defines the rules governing the derivation of domain information models from the RIM and the refinement of those models into HL7 standard specifications. The rules require that all information structures in derived models be traceable back to the RIM and that their semantic and related business rules not conflict with those specified in the RIM. The RIM therefore is the ultimate source for all information content in HL7 V3 standards.

The RIM is used by HL7 international affiliates to extend HL7 V3 standards to meet local needs. Through a process known as localization, V3 standard specifications are extended using the RIM as the source for new information content. This new information is derived from the RIM and refined in the same manner used to create the original specification.

0.1.3 Uses of the RIM Outside of HL7

The RIM is primarily for use by HL7 and its international affiliates. However, others outside of HL7 have also found the RIM useful. Although HL7 maintains a copyright on the expression of this standard, HL7 does not seek to license or otherwise control the use of information structures or programs that implement this specification. Early adopters of the V3 standards development process have used the RIM to develop HL7-like message specifications in their own environments. These early adopters include vendors, large integrated delivery networks, and government agencies within the United States and internationally. These same early adopters are extremely active in HL7 and provide practical input to the RIM and other aspects of V3 the development process.

Some HL7 member organizations have reported using the RIM as a source of input to their enterprise information architectures or as a starting place for systems analysis and design. The RIM may indeed be useful for such purposes; however, HL7 provides no assurance that the RIM is useful for anything other than as a reference model for HL7 standards development.

The RIM is only one model of healthcare information needs. The abstract style of the RIM and the ability to extend the RIM through vocabulary specifications make the RIM applicable to any conceivable healthcare system information interchange scenario. In fact, it is conceptually applicable to any information domain involving entities playing roles and participating in acts.

The universal applicability of the RIM makes it particularly useful for an organization like HL7 that has to consider the needs of a large and diverse membership. The style of the RIM makes it extremely stable, which is another important characteristic for HL7. The HL7 standards development process calls for the creation of domain specific models derived from the RIM and the incremental refinement of those models into design models that are specific to the problem area. These problem area specific design models narrow the abstractness of the RIM and include constraints on attribute values and class relationships that are use case specific. External organizations considering using the HL7 RIM are advised to adopt a similar process of deriving design models as a transformation of the RIM.

0.2 Further information

Questions or comments about the content of the standard may be addressed to HL7 at (www.hl7.org), to one of the HL7 International Affiliate organizations, or to the Secretariat of ISO TC215 – Health Informatics.

Health informatics — HL7 version 3 — Reference information model — Release 1

1 Scope

The Health Level Seven (HL7) Reference Information Model (RIM) is a static model of health and health care information as viewed within the scope of HL7 standards development activities. It is the combined consensus view of information from the perspective of the HL7 working group and the HL7 international affiliates. The RIM is the ultimate source from which all HL7 version 3.0 protocol specification standards draw their information-related content. In the context of ISO TC215 – Health Informatics, the RIM provides a reference model that may be used in developing further health informatics specifications.

2 Conformance

An information model such as the RIM specified in this standard may serve as the basis for other information models that are directly derived from it, and may provide a foundation to support the design of data bases and other information structures. Nevertheless, neither ISO TC 215 nor HL7 believe that it is reasonable to define tests of whether a particular implementation may conform to this standard. Therefore users of this standard shall not claim conformance to this standard. Further, ISO TC215 and HL7, as developers of this standard request that users inform them of particular requirements which caused the users to deviate from this stanard or to extend it. This will allow subsequent releases of the standard to meet a broader range of requirements.

3 Normative references

The following referenced documents are indispensable for the application 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 17113, *Health informatics — Exchange of information between healthcare information systems — Method for development of messages*

ISO/IEC 19501, *Information technology — Open Distributed Processing — Unified Modeling Language (UML) Version 1.4.2*

ANSI/HL7 V3 DT, R1-2004, HL7 Version 3 Standard: Data Types — Abstract Specification, Release 1

4 Terms and definitions

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

4.1
ANSI
American National Standards Institute

4.2
association
A reference from one class to another class or to itself, or a connection between two objects (instances of classes).

4.3
association role name
A name for each end of an association. The name is a short verb phrase depicting the role of the class at the opposite end of the association from the perspective of the class adjacent to the role.

4.4
attribute
An abstraction of a particular aspect of a class. Attributes become the data values that are passed in HL7 messages.

4.5
bag
A form of collection whose members are unordered, and need not be unique.

4.6
cardinality
Property of a data element (the number of times a data element may repeat within an individual occurrence of an object view).

4.7
class
An abstraction of a thing or concept in a particular application domain.

4.8
classifier attribute
An attribute used in generalization hierarchies to indicate which of the specializations is the focus of the class .

4.9
coded attribute
An attribute in the Reference Information Model (RIM) with a base data type of CD, CE, CS, or CV.

4.10
coding strength
An extensibility qualifier that specifies whether or not a code set can be expanded to meet local implementation needs.

4.11
coding system
A scheme for representing concepts using (usually) short concept identifiers to denote the concepts that are members of the system; defines a set of unique concept codes. Examples of coding systems are ICD-9, LOINC and SNOMED.

4.12
collection
An aggregation of similar objects. The forms of collection used by HL7 are set , bag, and list. Objects which may be found in collections include data types.

4.13
connection
In an information model, a specified relationship between two classes .

4.14
data type
The structural format of the data carried in an attribute. It may constrain the set of values an attribute may assume.

4.15
distal class
From the perspective of a class in an information model, it is the class at the opposite end of an association between the two.

4.16
domain
1. A particular area of interest. For example, the domain for HL7 is healthcare.

2. The set of possible values of a data type , attribute, or data type component. See also vocabulary domain.

4.17
event
1. A stimulus that causes a noteworthy change in the state of an object, or a signal that invokes the behavior of an object.

2. A vocabulary domain value for Mood.

4.18
extensibility qualifier
A vocabulary domain qualifier used in a domain specification, which indicates whether or not the existing vocabulary domain can be extended with additional values. There are two possible values: CNE (coded, no extension) and CWE (coded with extension).

4.19
generalization
An association between two classes, referred to as superclass and subclass, in which the subclass is derived from the superclass. The subclass inherits all properties from the superclass, including attributes, relationships, and states, but also adds new ones to extend the capabilities of the parent class. Essentially, a specialization from the point-of-view of the subclass.

4.20
generalization hierarchy
All superclasses and subclasses with a common root superclass.

4.21
graphical expression
A visual representation of a model that uses graphic symbols to represent the components of the model and the relationships that exist between those components.

4.22
Health Level Seven
HL7
A standards developing organization based in the United States of America.

4.23
identifier attribute
An attribute used to identify an instance of a class.

4.24
information model
A structured specification, expressed graphically and/or narratively, of the information requirements of a domain. An information model describes the classes of information required and the properties of those classes, including attributes, relationships, and states. Examples in HL7 are the Domain Reference Information Model, Reference Information Model, and Refined Message Information Model.

4.25
inheritance
In a generalization relationship, the subclass inherits all properties from the superclass, including attributes, relationships, and states, unless otherwise specified.

4.26
instance
A case or an occurrence. For example, an instance of a class is an object.

4.27
list
A form of collection whose members are ordered, and need not be unique.

4.28
literary expression
A representation of a model in text. The literary expression seeks to balance the need for a rigorous, unambiguous description of the model with the need for a rendition that can be easily read and interpreted by individuals who understand the general concepts underlying object-oriented models, but who may not be schooled in formal model definition languages.

4.29
mandatory
If an attribute is designated as mandatory, all message elements which make use of this attribute must contain a non-null value or they must have a default that is not null.

4.30
mandatory association
An association with a multiplicity minimum greater than zero on one end. A fully mandatory association is one with a multiplicity minimum greater than zero on both ends.

4.31
methodology
Methods or rules followed in a particular discipline.

4.32
model
A representation of a domain that uses abstraction to express the relevant concepts.

4.33
multiplicity
In the information model, multiplicity is a specification of the minimum and maximum number of objects from each class that can participate in an association. Multiplicity is specified for each end of the association.

4.34
namespace
A namespace is a part of the model in which names are defined and used, where each name has a unique meaning.

4.35
object
An instance of a class. A part of an information system containing a collection of related data (in the form of attributes) and procedures (methods) for operating on that data.

4.36
object identity
The feature that the existence of an object is independent of any values associated with the object.

4.37
object-based
Any method, language, or system that supports object identity, classification, and encapsulation. An object-based system does not support specialization . Ada is an example of an object-based implementation language.

4.38
property
Any attribute, association, method, or state model defined for a class or object.

4.39
Reference Information Model
RIM
The HL7 information model from which all other information models (e.g., R-MIMs) and messages are derived.

4.40
role
1. A function or position.

2. A Reference Information Model class that defines the competency of an Entity class. Each role is played by one Entity (the Entity that is in the role) and is usually scoped by another.

3. In UML, each end of an association is designated as a role to reflect the function that class plays in the association.

4.41
role name
See association role name.

4.42
set
A form of collection which contains an unordered list of unique elements of a single type.

4.43
specialization
An association between two classes (designated superclass and subclass), in which the subclass is derived from the superclass. The subclass inherits all properties from the superclass, including attributes, relationships, and states, but also adds new ones to extend the capabilities of the superclass.

4.44
state
A named condition of a class instance (object) that can be tested by examination of the instance's attributes and associations.

4.45
state attribute
An attribute describing the current state of an object.

4.46
state diagram
A graphical representation of a state transition model showing states as vertices (nodes) and state transitions as directed arcs (arrows) between the nodes.

4.47
state machine
A description of the life cycle for instances of a class, defined by a state transition model.

4.48
state transition
A change in the state of an object, as a result of a change in its attributes or associations.

4.49
state transition model
A graphical representation of the life cycle of a class. The model depicts all of the relevant states of a class, and the valid transitions from state to state.

4.50
subclass
A class that is the specialization of another class (superclass).

4.51
subject area
A convenient aggregation of model classes used to partition large models into manageable subsets.

4.52
sub-state
An identifiable state of a class that has a more specific definition than, and is entirely encompassed within the scope of, its super-state.

4.53
superclass
A class that is the generalization of one or more other classes (subclasses).

4.54
super-state
A state of a class that encompasses two or more independent sub-states.

4.55
Unified Modeling Language (UML)
A language for the creation of domain models. UML was created in order to unify several well-known object-oriented modeling methodologies, including those of Booch, Rumbaugh, Jacobson, and others.

4.56
vocabulary
The set of valid values for a coded attribute or field.

4.57
vocabulary domain
The set of all concepts that can be taken as valid values in an instance of a coded attribute or field; a constraint applicable to code values.

4.58
vocabulary domain qualifier
Part of a vocabulary domain specification. The two existing qualifiers are extensibility and realm.

4.59
W3C
The World Wide Web Consortium, an international industry consortium

5 Interpretation of the Specification

5.1 Specification contents

The RIM consists of classes assigned to one or more subject area packages. Attributes, Relationships, and State Machines are associated with classes.

Each class within the RIM represents information about a concept that must be documented and communicated within the health care environment. The names that are assigned to these classes are drawn from normal language, but the use of these names is necessarily constrained to the "namespace" of the RIM. The meaning of these classes is entirely embodied in the definition of the class, and the definitions of the properties (attributes and associations) assigned to that class. Thus, for example, the meaning of the "Role" class can only be understood by studying the definition provided and the properties assigned. Definitions from another context or dictionary definitions for the name are not relevant within the context of the RIM namespace.

The RIM is expressed using the Unified Modeling Language (UML) with HL7 specific tags as extensions to the UML model element metadata. All standard UML model element metadata values are normative but only the following HL7 extensions are also normative:

- Class.stateAttribute
- Class.classCode

- Attribute.mandatoryInclusion
- Attribute.cardinality
- Attribute.vocabDomain
- Attribute.vocabStrength

5.2 Understanding the RIM

The RIM uses a very abstract modeling style. The back-bone classes of the RIM and their structural attributes are its core. An understanding of these classes and attributes is essential to understanding the RIM. This section describes how the abstractions are represented in UML and controlled through the application of controlling vocabulary that is part of this specification. An "executive overview" or high-level tutorial that provides examples of how these abstractions can be used to represent more detailed health information is contained in Annex B.

5.2.1 RIM as an abstract model

The RIM is comprised of six "back-bone" classes:

- Act which represents the actions that are executed and must be documented as health care is managed and provided;
- Participation which expresses the context for an act in terms such as who performed it, for whom it was done, where it was done, etc.;
- Entity which represents the physical things and beings that are of interest to, and take part in health care;
- Role which establishes the roles that entities play as they participate in health care acts;
- ActRelationship which represents the binding of one act to another, such as the relationship between an order for an observation and the observation event as it occurs; and
- RoleLink which represents relationships between individual roles.

Three of these classes -- Act, Entity and Role -- are further represented by a set of specialized classes, or sub-types. In the HL7 representation, a sub-type is only added to the RIM if it requires one or more attributes or associations that are not inherited from its parents. Classes that represent distinct concepts, but which need no further attributes or associations are represented solely as a unique code in the controlling vocabulary. Therefore, these three classes include the following coded attributes, which serve to further define the concept being modeled:

- **classCode** (in Act, Entity and Role) represents the exact class or concept intended, whether or not that class is represented as a class in the RIM hierarchy;
- **moodCode** (in Act) and **determinerCode** (in Entity) an attribute that distinguishes whether the class represents an instance or a kind of Act or Entity. If the class is a specialization of Act then moodCode further delineates the instance as an occurrence or an intent;
- **code** (in Act, Entity and Role) provides for further classification within a particular classCode value, such as a particular type of observation within the Observation class.

The other three RIM back-bone classes -- Participation, ActRelationship and RoleLink -- are not represented by generalization-specialization hierarchies. Nevertheless, these classes represent a variety of concepts, such as different forms of participation or different kinds of relationships between acts. These distinctions are represented by a **typeCode** attribute that is asserted for each of these classes.

5.2.2 Representation of RIM class structure

As noted previously, the RIM is modeled using a subset of the semantics embodied in UML. The RIM is a set of UML classes, each containing one or more attributes, which are assigned a data type based on an independent specification of data types. The classes are linked either by a set of association relationships, identified by unique role names, or by generalization relationships.

Each of these elements includes a textual definition. The appearance of attributes and associations is controlled by cardinality and related constraints applied to the attributes and to the roles that link the associations to the classes.

5.2.3 Representation of controlling vocabulary

Several of the attributes in the RIM are coded with the CS data type, which means that the value set to be used for those attributes must be drawn from the set of codes that have been defined by HL7. The classCode and typeCode attributes mentioned above are examples of attributes with a CS data type.

All of the code sets for these attributes are included as part of this ballot and are subject to the same voting principles as the classes in the RIM. Each code set is represented as a vocabulary domain -- the set of all concepts that can be taken as valid values in an instance of a coded field or attribute. It is important to note that a vocabulary domain consists of a set of concepts, not a set of words or codes.

5.2.4 Related specifications

As noted above, each attribute in the RIM is assigned a data type. The formal specification for these data types differs depending upon whether this model is being used within the context of HL7 or the context of ISO TC215. For use in HL7, the normative specification for the datatypes is the "HL7 Data Types Abstract Specification ((need current name)). Within the ISO TC215 framework, work-items to specify a relevant data type specification are being advanced as ISO 22??xx??. In either case, an informative summary of the relevant data type properties is included as Annex C.

5.3 Graphic Diagrams of the RIM

The classes in the Normative content of the RIM are represented in the following diagrams:

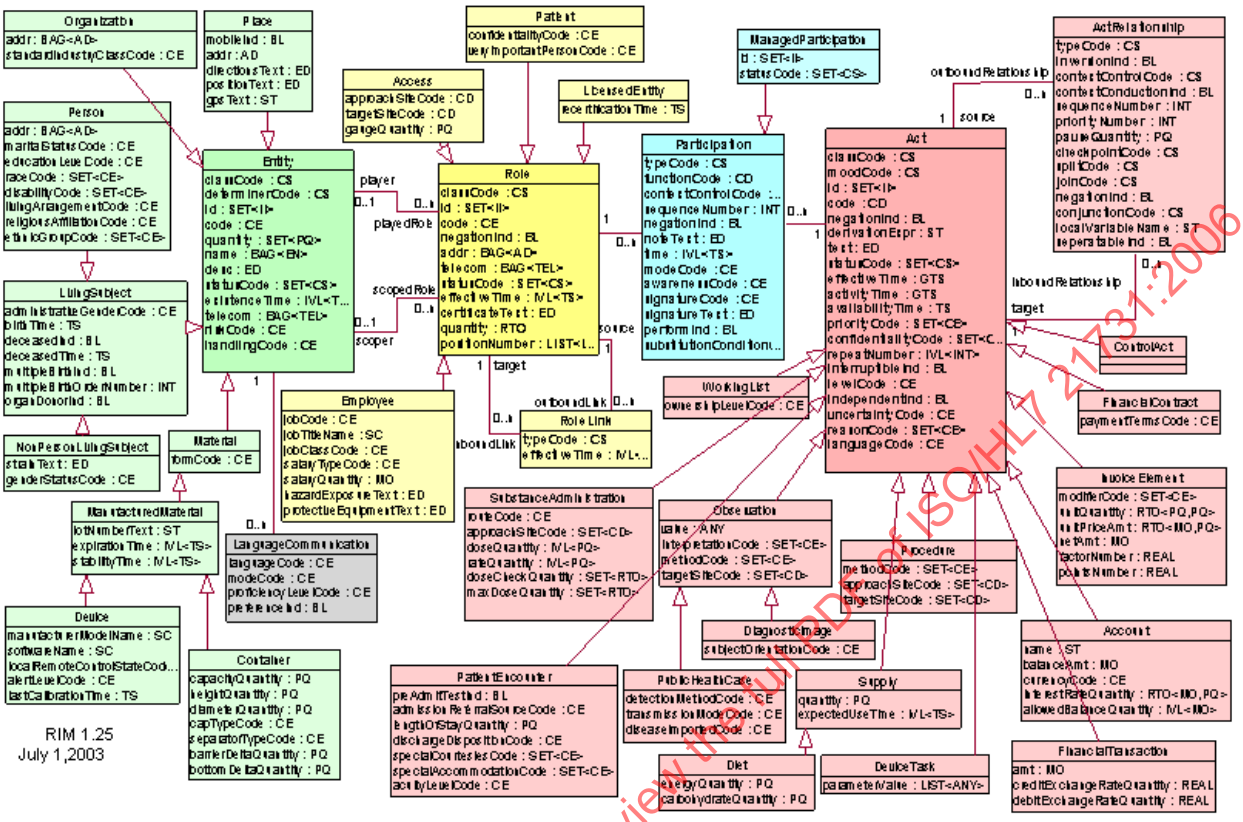

Figure 1: UML diagram of the classes in all subject areas.

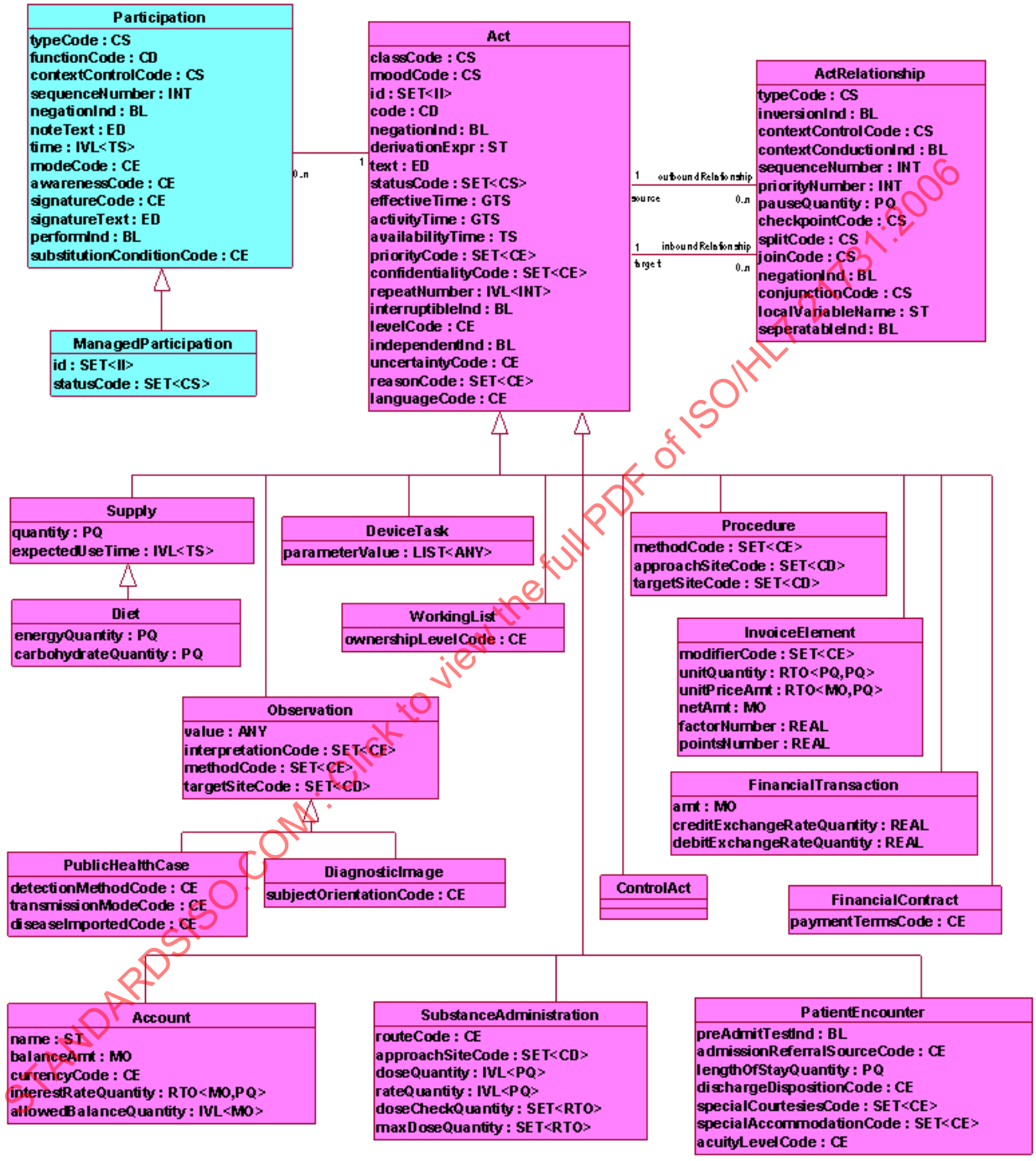

Figure 2: UML diagram of the classes in the Acts subject area.

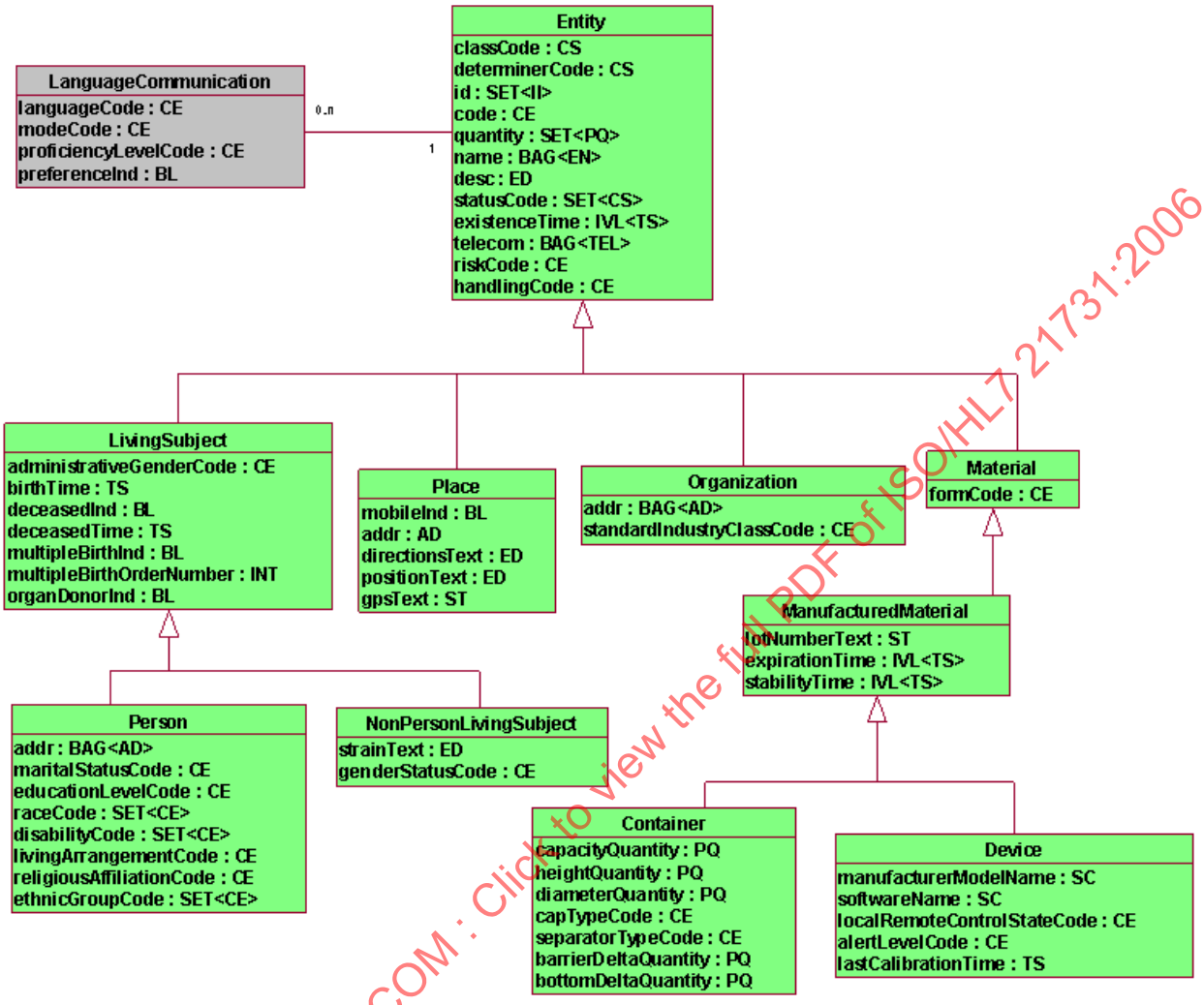

Figure 3: UML diagram of the classes in the Entities subject area.

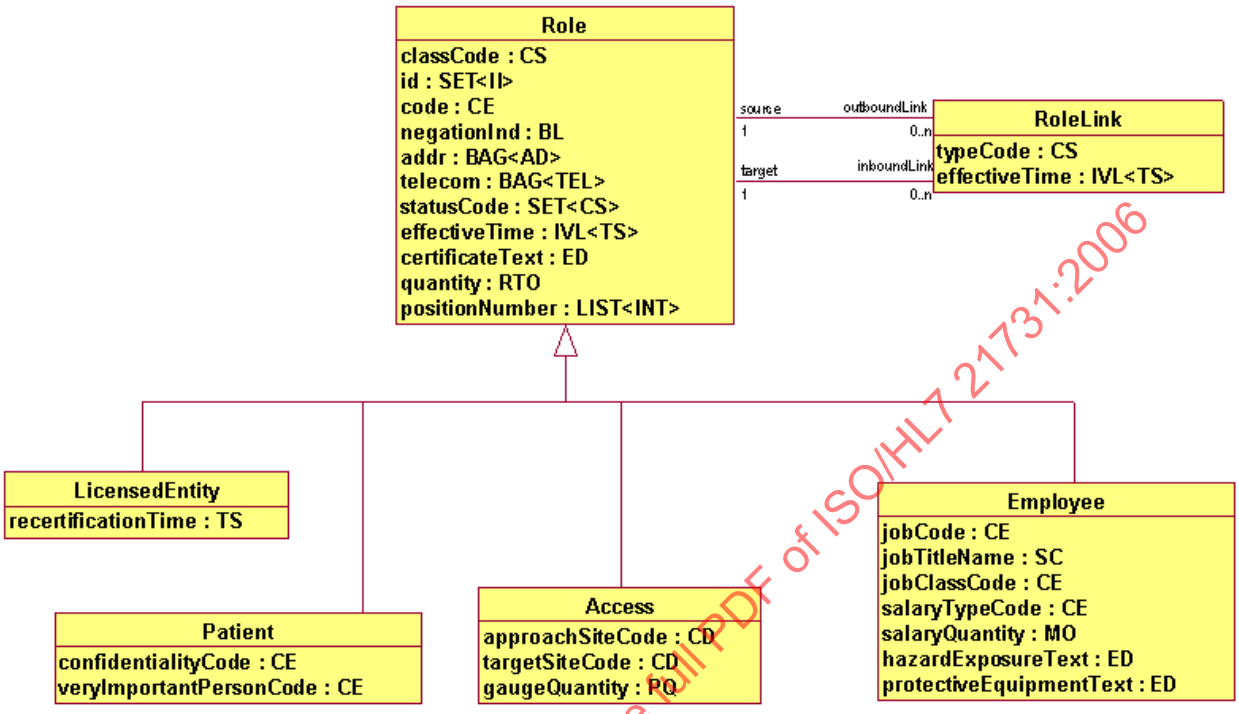

Figure 4: UML diagram of the classes in the Roles subject area.

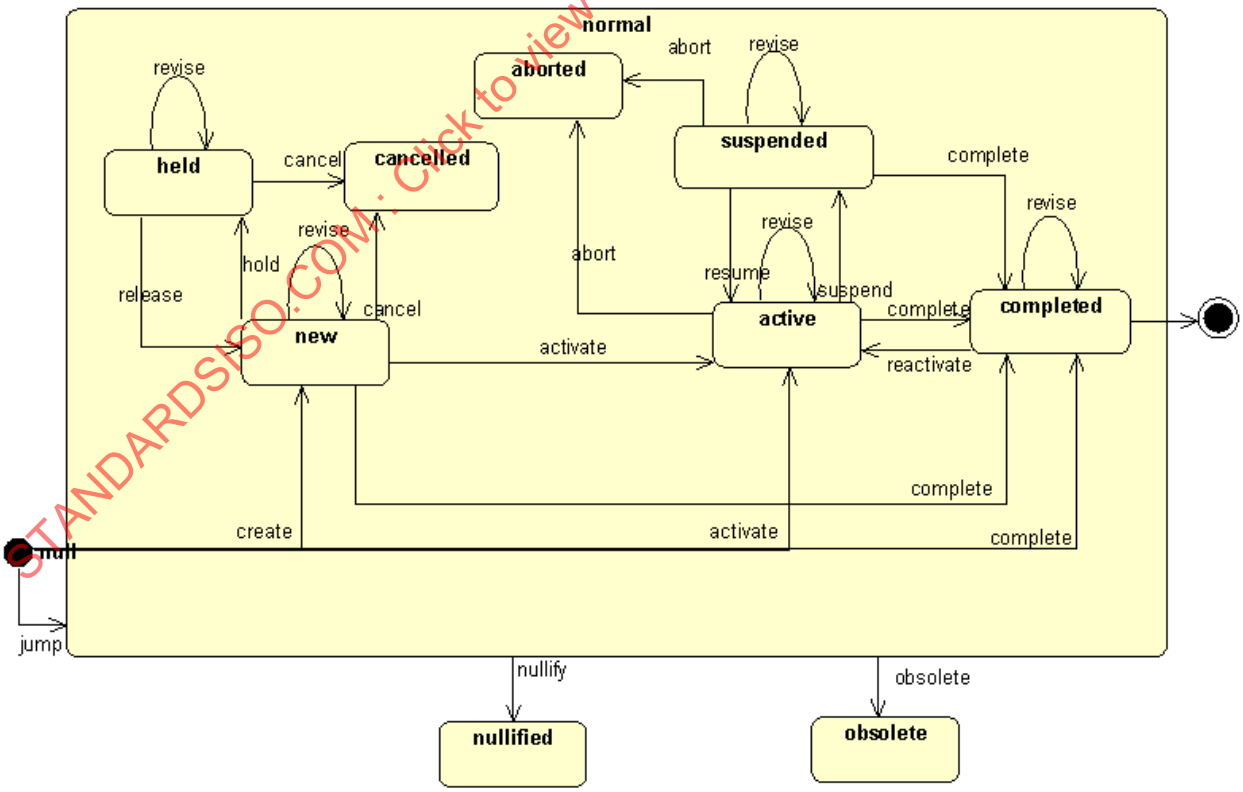

Figure 5: State-machine diagram for Act class.

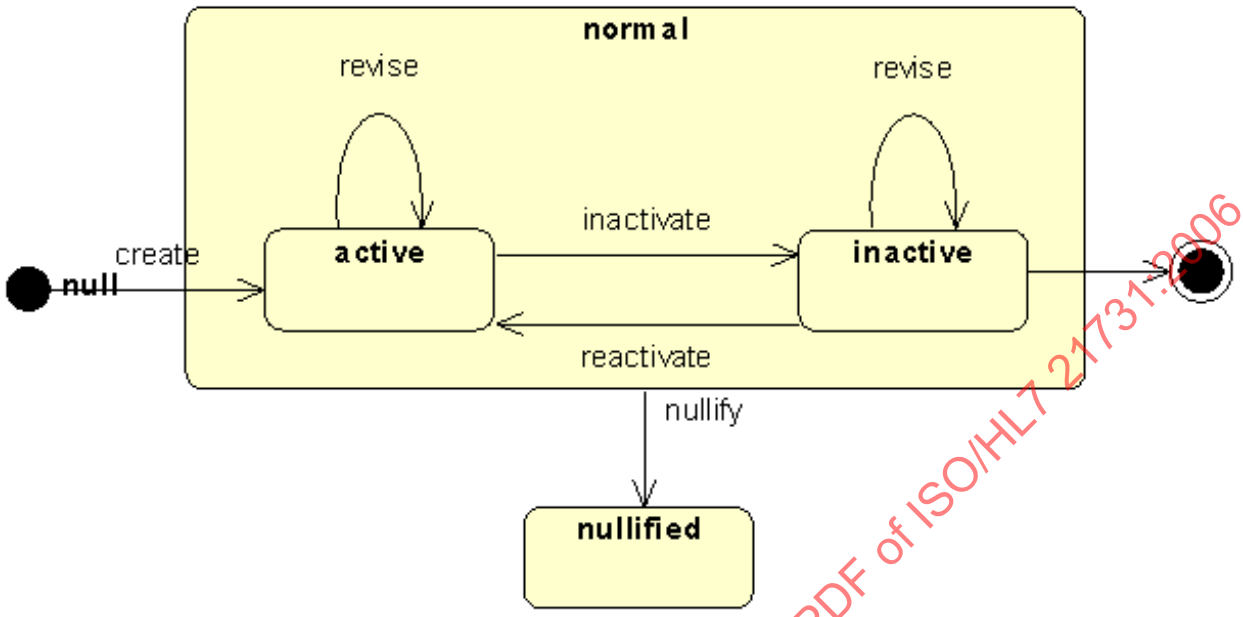

Figure 6: State-machine diagram for Entity class.

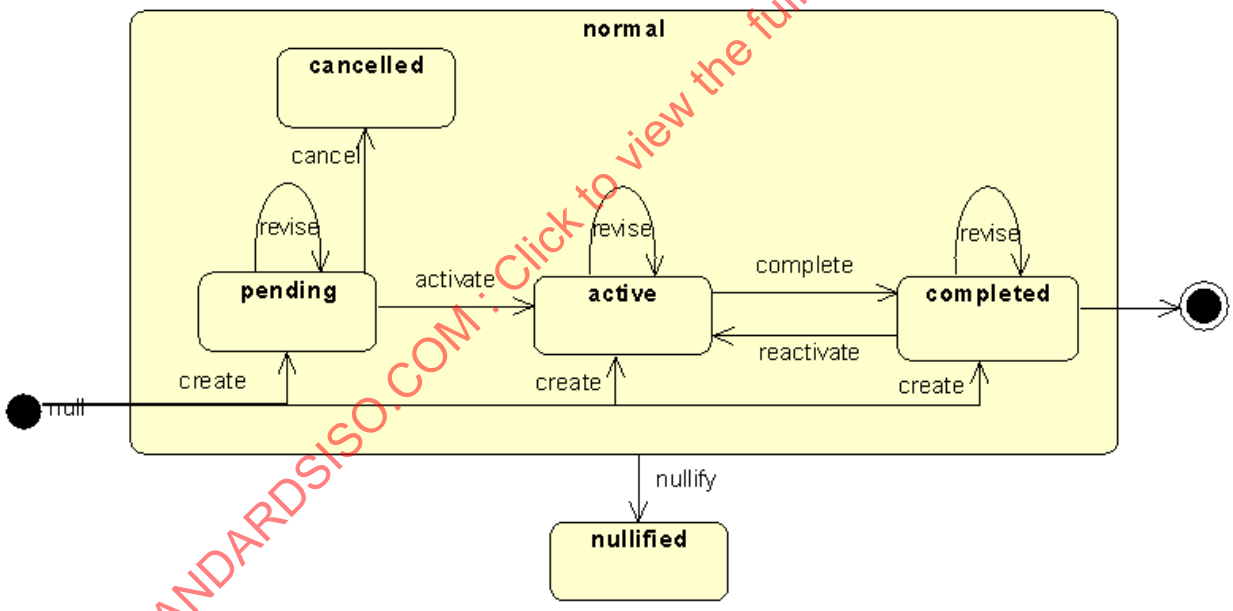

Figure 7: State-machine diagram for ManagedParticipation class.

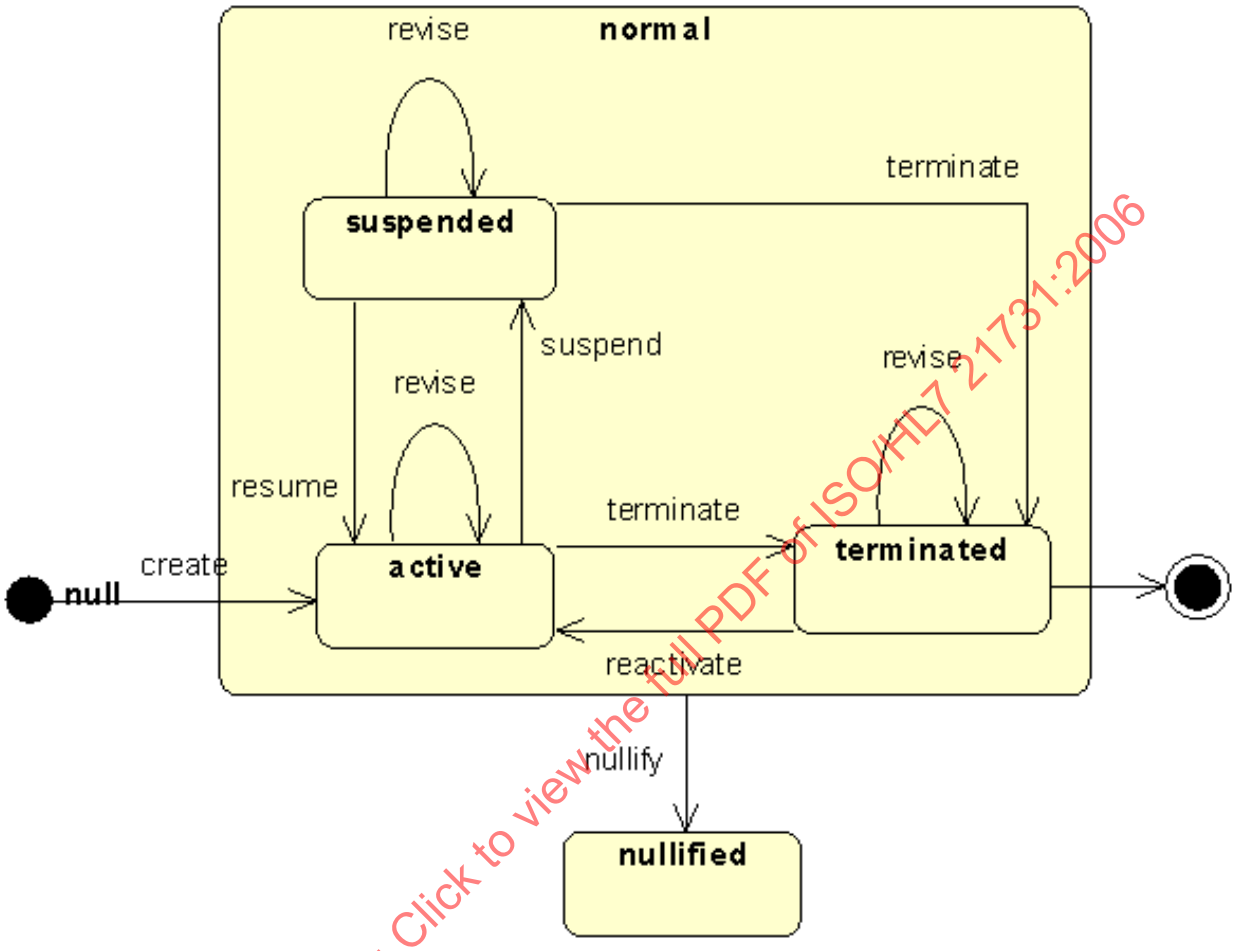

Figure 8: State-machine diagram for Role class.

6 SUBJECT AREAS IN: HEALTH LEVEL SEVEN REFERENCE INFORMATION MODEL, RELEASE 1

FoundationClasses

6.1 FoundationClasses (in base model)

This collection of classes and their associations represent the "normative" content of the HL7 RIM. The content of this subject area has been balloted within HL7 as a normative document.

Link to a class diagram of this subject area.

- FoundationClasses contains subject areas:

Acts Entities Roles

6.1.1 Acts (in FoundationClasses)

A collection of classes including the Act class and its specializations. These relate to the actions and events that constitute health care services.

Link to a class diagram of this subject area.

- Acts contains classes:

Account	FinancialContract	PatientEncounter
Act	FinancialTransaction	Procedure
ActRelationship	InvoiceElement	PublicHealthCase
DeviceTask	ManagedParticipation	SubstanceAdministration
DiagnosticImage	Observation	Supply
Diet	Participation	WorkingList

6.1.2 Entities (in FoundationClasses)

A collection of classes related to the Entity class, its specializations and related qualifying classes. The classes represent health care stakeholders and other things of interest to health care.

Link to a class diagram of this subject area.

- Entities contains classes:

Container	LivingSubject	Organization
Device	ManufacturedMaterial	Person
Entity	Material	Place
LanguageCommunication	NonPersonLivingSubject	

6.1.3 Roles (in FoundationClasses)

A collection of classes related to the Role class and its specializations. These classes focus on the roles participants may play in health care.

Link to a class diagram of this subject area.

- Roles contains classes:

Access	LicensedEntity	Role
Employee	Patient	RoleLink

7 Classes in: Health Level Seven Reference Information Model, Release 1

Each of the classes is listed below. The sort order is based on the following three criteria:

1. Normative Content first
2. Primary subject area name, alphabetically
3. Class name, alphabetically

7.1 Classes in subject area Acts

7.1.1 Class: Act *(in Acts)*

- Class code: ACT
- Attributes of Act:

classCode :: CS	statusCode :: SET<CS>	interruptibleInd :: BL
moodCode :: CS	effectiveTime :: GTS	levelCode :: CE
id :: SET<II>	activityTime :: GTS	independentInd :: BL
code :: CD	availabilityTime :: TS	uncertaintyCode :: CE

negationInd :: BL	priorityCode :: SET<CE>	reasonCode :: SET<CE>
derivationExpr :: ST	confidentialityCode :: SET<CE>	languageCode :: CE
text :: ED	repeatNumber :: IVL<INT>	

- Associations of Act:

  outboundRelationship :: (0..*) ActRelationship :: **source** :: (1..1)
  inboundRelationship :: (0..*) ActRelationship :: **target** :: (1..1)
  participation :: (0..*) Participation :: **act** :: (1..1)

- Act generalizes:

Account	InvoiceElement	SubstanceAdministration
DeviceTask	Observation	Supply
FinancialContract	PatientEncounter	WorkingList
FinancialTransaction	Procedure	

Description of Act:

A record of something that is being done, has been done, can be done, or is intended or requested to be done.

Examples: The kinds of acts that are common in health care are (1) a clinical observation, (2) an assessment of health condition (such as problems and diagnoses), (3) healthcare goals, (4) treatment services (such as medication, surgery, physical and psychological therapy), (5) assisting, monitoring or attending, (6) training and education services to patients and their next of kin, (7) and notary services (such as advanced directives or living will), (8) editing and maintaining documents, and many others.

Discussion and Rationale: Acts are the pivot of the RIM; all domain information and processes are represented primarily in Acts. Any profession or business, including healthcare, is primarily constituted of intentional actions, performed and recorded by responsible actors. An Act-instance is a record of such an intentional action. Intentional actions are distinguished from something that happens by forces of nature (natural events). Such natural events are not Acts by themselves, but may be recorded as observed (Observation).

Acts connect to Entities in their Roles through Participations and connect to other Acts through ActRelationships. Participations are the authors, performers and other responsible parties as well as subjects and beneficiaries (which includes tools and material used in the performance of the act, which are also subjects). The moodCode distinguishes between Acts that are meant as factual records, vs. records of intended or ordered services, and the other modalities in which act can appear.

One of the Participations that all acts have (at least implicitly) is a primary author, who is responsible of the Act and who "owns" the act. Responsibility for the act means responsibility for what is being stated in the Act and as what it is stated. Ownership of the act is assumed in the sense of who may operationally modify the same act. Ownership and responsibility of the Act is not the same as ownership or responsibility of what the Act-object refers to in the real world. The same real world activity can be described by two people, each being the author of their Act, describing the same real world activity. Yet one can be a witness while the other can be a principal performer. The performer has responsibilities for the physical actions; the witness only has responsibility for making a true statement to the best of his or her ability. The two Act-instances may even disagree, but because each is properly attributed to its author, such disagreements can exist side by side and left to arbitration by a recipient of these Act-instances.

In this sense, an Act-instance represents a "statement" according to Rector and Nowlan (1991) [Foundations for an electronic medical record. Methods Inf Med. 30.] Rector and Nowlan have emphasized the importance of understanding the medical record not as a collection of facts, but "a faithful record of what clinicians have heard, seen, thought, and done." Rector and Nowlan go on saying that "the other requirements for a medical record, e.g., that it be attributable and permanent, follow naturally from this view." Indeed the Act class is this attributable statement, and the rules of updating acts (discussed in the state-transition model, see Act.statusCode) versus generating new Act-instances are designed according to this principle of permanent attributable statements.

Rector and Nolan focus on the electronic medical record as a collection of statements, while attributed statements, these are still mostly factual statements. However, the Act class goes beyond this limitation to attributed factual statements, representing what is known as "speech-acts" in linguistics and philosophy. The notion of speech-act includes that there is pragmatic meaning in language utterances, aside from just factual statements; and that these utterances interact with the real world to change the state of affairs, even directly cause physical activities to happen. For example, an order is a speech act that (provided it is issued adequately) will cause the ordered action to be physically performed. The speech act theory has culminated in the seminal work by Austin (1962) [How to do things with words. Oxford University Press].

An activity in the real world may progress from defined, through planned and ordered to executed, which is represented as the mood of the Act. Even though one might think of a single activity as progressing from planned to executed, this progression is reflected by multiple Act-instances, each having one and only one mood that will not change along the Act-instance's life cycle. This is because the attribution and content of speech acts along this progression of an activity may be different, and it is often critical that a permanent and faithful record be maintained of this progression. The specification of orders or promises or plans must not be overwritten by the specification of what was actually done, so as to allow comparing actions with their earlier specifications. Act-instances that describe this progression of the same real world activity are linked through the ActRelationships (of the relationship category "sequel").

Act as statements or speech-acts are the only representation of real world facts or processes in the HL7 RIM. The truth about the real world is constructed through a combination (and arbitration) of such attributed statements only, and there is no class in the RIM whose objects represent "objective state of affairs" or "real processes" independent from attributed statements. As such, there is no distinction between an activity and its documentation. Every Act includes both to varying degrees. For example, a factual statement made about recent (but past) activities, authored (and signed) by the performer of such activities, is commonly known as a procedure report or original documentations (e.g., surgical procedure report, clinic note etc.). Conversely, a status update on an activity that is presently in progress, authored by the performer (or a close observer) is considered to capture that activity (and is later superceded by a full procedure report). However, both status update and procedure report are acts of the same kind, only distinguished by mood and state (see statusCode) and completeness of the information.

Attributes of Act:

7.1.1.1 Act.**classCode :: CS** (1..1) **Mandatory**

Vocabulary domain: ActClass (CNE)

Attribute description:
A code specifying the major type of Act that this Act-instance represents.

Constraints: The classCode domain is a tightly controlled vocabulary, not an external or user-defined vocabulary.

Every Act-instance must have a classCode. If the act class is not further specified, the most general Act.classCode (ACT) is used.

The Act.classCode must be a generalization of the specific Act concept (e.g., as expressed in Act.code), in other words, the Act concepts conveyed in an Act must be specializations of the Act.classCode. Especially, the classCode is not a "modifier" or the Act.code that can alter the meaning of a class code. (See Act.code for contrast.)

7.1.1.2 Act.moodCode :: CS (1..1) Mandatory

Vocabulary domain: ActMood (CNE)

Attribute description:
A code distinguishing whether an Act is conceived of as a factual statement or in some other manner as a command, possibility, goal, etc.

Constraints: An Act-instance must have one and only one moodCode value.

The moodCode of a single Act-instance never changes. Mood is not state.

To describe the progression of a business activity from defined to planned to executed, etc. one must instantiate different Act-instances in the different moods and link them using ActRelationship of general type "sequel". (See ActRelationship.typeCode.)

Discussion: The Act.moodCode includes the following notions: (1) event, i.e., factual description of an actions that occurred; (2) definition of possible actions and action plans (the master file layer); (3) intent, i.e., an action plan instantiated for a patient as a care plan or order; (4) goal, i.e., an desired outcome attached to patient problems and plans; and (5) criterion, i.e., a predicate used to evaluate a logical expression.

The Act.moodCode modifies the meaning of the Act class in a controlled way, just as in natural language, grammatical form of a verb modify the meaning of a sentence in defined ways. For example, if the mood is factual (event), then the entire act object represents a known fact. If the mood expresses a plan (intent), the entire act object represents the expectation of what should be done. The mood does not change the meaning of individual act properties in peculiar ways.

Since the mood code is a determining factor for the meaning of an entire Act object, the mood must always be known. This means, whenever an act object is instantiated, the mood attribute must be assigned to a valid code, and the mood assignment cannot change throughout the lifetime of an act object.

As the meaning of an act object is factored in the mood code, the mood code affects the interpretation of the entire Act object and with it every property (attributes and associations). Note that the mood code affects the interpretation of the act object, and the meaning of the act object in turn determines the meaning of the attributes. However, the mood code does not arbitrarily change the meaning of individual attributes.

Inert vs. descriptive properties of Acts: Acts have two kinds of act properties, inert and descriptive properties. Inert properties are not affected by the mood, descriptive properties follow the mood of the object. For example, there is an identifier attribute Act.id, which gives a unique identification to an act object. Being a unique identifier for the object is in no way dependent on the mood of the act object. Therefore, the "interpretation" of the Act.id attribute is inert with respect to the act object's mood.

By contrast, most of the Act class' attributes are descriptive for what the Act statement expresses. Descriptive properties of the Act class give answer to the questions who, whom, where, with what, how and when the action is done. The questions who, whom, with what, and where are answered by Participations, while how

and when are answered by descriptive attributes and ActRelationships. The interpretation of a descriptive attribute is aligned to the interpretation of the entire act object, and controlled by the mood.

Examples: To illustrate the effect of mood code, consider a "blood glucose" observation:

The DEFINITION mood specifies the Act of "obtaining blood glucose". Participations describe in general the characteristics of the people who must be involved in the act, and the required objects, e.g., specimen, facility, equipment, etc. involved. The Observation.value specifies the absolute domain (range) of the observation (e.g., 15-500 mg/dl).

In INTENT mood the author of the intent expresses the intent that he or someone else "should obtain blood glucose". The participations are the people actually or supposedly involved in the intended act, especially the author of the intent or any individual assignments for group intents, and the objects actually or supposedly involved in the act (e.g., specimen sent, equipment requirements, etc.). The Observation.value is usually not specified, since the intent is to measure blood glucose, not to measure blood glucose in a specific range. (But compare with GOAL below).

In REQUEST mood, a kind of intent, the author requests to "please obtain blood glucose". The Participations are the people actually and supposedly involved in the act, especially the placer and the designated filler, and the objects actually or supposedly involved in the act (e.g., specimen sent, equipment requirements, etc.). The Observation.value is usually not specified, since the order is not to measure blood glucose in a specific range.

In EVENT mood, the author states that "blood glucose was obtained". Participations are the people actually involved in the act, and the objects actually involved (e.g., specimen, facilities, equipment). The Observation.value is the value actually obtained (e.g., 80 mg/dL, or <15 mg/dL).

In event-CRITERION mood, an author considers a certain class of "obtaining blood glucose" possibly with a certain value (range) as outcome. The Participations constrain the criterion, for instance, to a particular patient. The Observation.value is the range in which the criterion would hold (e.g. > 180 mg/dL or 200-300 mg/dL).

In GOAL mood (a kind of criterion), the author states that "our goal is to be able to obtain blood glucose with the given value (range)". The Participations are similar to intents, especially the author of the goal and the patient for whom the goal is made. The Observation.value is the range which defined when the goal is met (e.g. 80-120 mg/dl).

Rationale: The notion of "mood" is borrowed from natural language grammar, the mood of a verb (lat. modus verbi).

The notion of mood also resembles the various extensions of the logic of facts in modal logic and logic with modalities, where the moodCode specifies the modality (fact, possibility, intention, goal, etc.) under which the Act-statement is judged as appropriate or defective.

7.1.1.3 Act**.id :: SET<II>** (0..*)

Attribute description:
A unique identifier for the Act.

7.1.1.4 Act**.code :: CD** (0..1)

Vocabulary domain: ActCode (CWE)

Attribute description:
A code specifying the particular kind of Act that the Act-instance represents within its class.

Constraints: The kind of Act (e.g. physical examination, serum potassium, inpatient encounter, charge financial transaction, etc.) is specified with a code from one of several, typically external, coding systems. The coding system will depend on the class of Act, such as LOINC for observations, etc.

Conceptually, the Act.code must be a specialization of the Act.classCode. This is why the structure of ActClass domain should be reflected in the superstructure of the ActCode domain and then individual codes or externally referenced vocabularies subordinated under these domains that reflect the ActClass structure.

Act.classCode and Act.code are not modifiers of each other but the Act.code concept should really imply the Act.classCode concept. For a negative example, it is not appropriate to use an Act.code "potassium" together with and Act.classCode for "laboratory observation" to somehow mean "potassium laboratory observation" and then use the same Act.code for "potassium" together with Act.classCode for "medication" to mean "substitution of potassium". This mutually modifying use of Act.code and Act.classCode is not permitted.

Discussion: Act.code is not a required attribute of Act. Rather than naming the kind of Act using an Act.code, one can specify the Act using only the class code and other attributes and properties of the Act. In general and more commonly, the kind of Act is readily specified by an ActRelationship specifying that this Act instantiates another Act in definition mood. Or, even without reference to an act definition, the act may be readily described by other attributes, ActRelationships and Participations. For example, the kind of SubstanceAdministration may be readily described by referring to the specific drug, as the Participation of an Entity representing that drug.

7.1.1.5 Act**.negationInd :: BL** (0..1)

Attribute description:
An indicator specifying that the Act statement is a negation of the Act as described by the descriptive attributes.

Examples: Used with an Observation event, it allows one to say "patient has NO chest pain". With an Observation criterion it negates the criterion analogously, e.g., "if patient has NO chest pain for 3 days ...", or "if systolic blood pressure is **not** within 90-100 mm Hg ..."

Discussion: The negationInd works as a negative existence quantifier. This is best explained on Acts in criterion mood, and then translates into all other moods. In criterion mood without negation, one usually only specifies a few critical attributes and relationships (features) of an Act, i.e., only those that are needed to test the criterion. The more features one specifies, the more constrained (specific) is the criterion. For example, to test for "systolic blood pressure of 90-100 mm Hg", one would use only the descriptive attributes Act.code (for systolic blood pressure) and Observation.value (for 90-100 mm Hg). If one would also specify an effectiveTime, i.e., for "yesterday", the criterion would be more constrained. If the negationInd is true for the above criterion, then the meaning of the test is whether a systolic blood pressure of 90-100 mm Hg yesterday does **not exist** (independent of whether any blood pressure was measured).

The negationInd negates the Act as described by the descriptive properties (including Act.code, Act.effectiveTime, Observation.value, Act.doseQty, etc.) and any of its components. The inert properties such as Act.id, Act.moodCode, Act.confidentialityCode, and particularly the Author-Participation are **not** negated. These inert properties always have the same meaning: i.e., the author remains to be the author of the negative observation. Also, most ActRelationships (except for components) are not included in the negation.

For example, a highly confidential order written by Dr. Jones, to explicitly **not** give "succinyl choline" for the "reason" (ActRelationship) of a history of malignant hyperthermia (Observation) negates the descriptive properties "give succinyl choline" (Act.code), but it is still positively an order and written by Dr. Jones and for patient John Smith, and the reason for this order is the patient's history of malignant hyperthermia.

However, additional detail in descriptive attributes will be part of the negation which then limits the effectiveness of the negated statement. For example, had the order not to give a substance included a doseQuantity, it would mean that the substance should not be given at that particular dose (but any other dose might still be O.K.).

An act statement with negationInd is still a statement about the specific fact described by the Act. For instance, a negated "finding of wheezing on July 1" means that the author positively denies that there was wheezing on July 1, and that he takes the same responsibility for such statement and the same requirement to have evidence for such statement than if he had not used negation. Conversely, negation indicator does **not** just negate that the fact was affirmed or that the statement had been made. This holds for all moods in the same way, e.g., a negated order is an order **not** to do the described act, not just the lapidary statement that there is no such order.

7.1.1.6 Act**.derivationExpr :: ST** (0..1)

Attribute description:
A character string containing a formal language expression that specifies how the Act's attributes are derived from input parameters associated with derivation relationships.

Discussion: Derived observations can be defined through association with other observations using ActRelationships of type "derivation". For example, to define a derived observation for Mean Corpuscular Hemoglobin (MCH) one will associate the MCH observation with a Hemoglobin (HGB) observation and a Red Blood cell Count (RBC) observation. And the derivation expression encodes the formula: MCH = HGB / RBC.

The derivation expression is represented as a character string.

[Note: The syntax of that expression is yet to be fully specified. There would be a single standard expression language rather than an optional choice between many expression languages. The syntax would be based on a de-facto standard for many object-oriented languages, such as C++, Java, OCL etc. A concrete specification of this expression language is being worked on now and drafts can be expected within the year 2003.]

7.1.1.7 Act**.text :: ED** (0..1)

Attribute description:
A textual or multimedia description of the Act.

Examples: For act definitions, the Act.text can contain textbook-like information about that act. For act orders, the description will contain particular instructions pertaining only to that order.

Constraints: No restriction on length or content is imposed on the Act.text attribute.

The content of the description is not considered part of the functional information communicated between computer systems.

Free text descriptions are used to help an individual interpret the content and context of the act, but all information relevant for automated functions must be communicated using the proper attributes and associated objects.

For Acts that involve human readers and performers, however, computer systems must show the Act.text field to a human user, who has responsibility for the activity; or at least must indicate the existence of the Act.text information and allow the user to see that information.

7.1.1.8 Act**.statusCode :: SET<CS>** (0..*)

Vocabulary domain: ActStatus (CNE)

Attribute description:
A code specifying the state of the Act.

7.1.1.9 Act**.effectiveTime :: GTS** (0..1)

Attribute description:
A time expression specifying the focal or operative time of the Act, the primary time for which the Act holds, the time of interest from the perspective of the Act's intention.

Examples: For clinical Observations, the effectiveTime is the time at which the observation holds (is effective) for the patient.

For contracts, the effectiveTime is the time for which the contract is in effect.

For consents, the effectiveTime is the time for which the consent is valid.

For substance administrations, the effective time is the time over which the substance is to be administered, including the frequency of administration (e.g. TID for 10 days)

For a surgical procedure (operation), the effectiveTime is the time relevant for the patient, i.e., between incision and last suture.

For transportation acts, the effective time is the time the transported payload is en route.

For patient encounters, this is the "administrative" time, i.e., the encounter start and end date required to be chosen by business rules, as opposed to the actual time the healthcare encounter related work is performed.

Discussion: The effectiveTime is also known as the "primary" time (Arden Syntax) or the "biologically relevant time" (HL7 v2.x). This attribute is distinguished from activityTime.

For observations, the time of the observation activity may be much later than the time of the observed feature. For instance, in a Blood Gas Analysis (BGA), a result will always come up several minutes after the specimen was taken, meanwhile the patient's physiological state may have changed significantly.

For essentially physical activities (surgical procedures, transportations, etc.), the effective time is the time of interest for the Act's intention, i.e., since the intention of a transportation is to deliver a payload from location A to B, the effectiveTime is the time this payload is underway from A to B. However, the Act usually also includes accidental work which is necessary to perform the intention of the Act, but is not relevant for the Act's purpose.

For example, the time a driver needs to go to the pick-up location A and then return from drop-off location B to some home base, is included in the physical activity, but does not matter from the perspective of the payload's transportation. Another example is: a person's work hours (effectiveTime) may be from 8 AM to 5 PM, no matter whether that person needs 10 minutes for the commute or 2 hours. The commute is necessary to be at work, but it is not essential for the work hours.

7.1.1.10 Act**.activityTime :: GTS** (0..1)

Attribute description:
A time expression specifying the time in which the activity referred to by the Act statement happened, or -- depending on the mood -- is ordered to happen, scheduled to happen, when it can possibly happen, etc.

Discussion: See also, effectiveTime. The activityTime is the total time of activity including preparation and clean-up actions that may not be considered relevant to the main action. Thus the activityTime may be longer than the effectiveTime of the same act or it may be entirely different. For example, in retrospective observations, the observation activityTime is much later than the effectiveTime for which the observation is made.

For most use cases, designers should first consider effectiveTime as the primary relevant time for an act.

7.1.1.11 Act**.availabilityTime :: TS** (0..1)

Attribute description:
The point in time at which information about Act-instance (regardless of mood) first became available to a system reproducing this Act.

Examples: An Act might record that a patient had a right-ventricular myocardial infarction effective three hours ago (see Act.effectiveTime), but we may only know about this unusual condition a few minutes ago (Act.availabilityTime). Thus, any interventions from three hours ago until a few minutes ago may have assumed the more common left-ventricular infarction, which can explain why these interventions (e.g., nitrate administration) may not have been appropriate in light of the more recent knowledge.

Discussion: The availabilityTime is a subjective secondary piece of information added (or changed) by a system that reproduces this Act, and is not attributed to the author of the act statement (it would not be included in the material the author would attest to with a signature). The system reproducing the Act is often not the same as the system originating the Act, but a system that received this Act from somewhere else, and, upon receipt of the Act, values the availabilityTime to convey the time since the users of that particular system could have known about this Act-instance.

When communicating availabilityTime to another system, the availabilityTime of an Act A is attributed to the author of another Act B, that refers to or includes A. For example, if a medical record extract is compiled for reporting adverse events, availabilityTimes are attributed to the author who compiles that report.

7.1.1.12 Act**.priorityCode :: SET<CE>** (0..*)

Vocabulary domain: ActPriority (CWE)

Attribute description:
A code or set of codes (e.g., for routine, emergency), specifying the urgency under which the Act happened, can happen, is happening, is intended to happen, or is requested/demanded to happen.

Discussion: This attribute is used in orders to indicate the ordered priority, and in event documentation it indicates the actual priority used to perform the act. In definition mood it indicates the available priorities.

7.1.1.13 Act**.confidentialityCode :: SET<CE>** (0..*)

Vocabulary domain: Confidentiality (CWE)

Attribute description:
A code that controls the disclosure of information about this Act, regardless of mood.

Discussion: It is important to note that the necessary confidentiality of the medical record cannot be achieved solely through confidentiality codes to mask individual record items from certain types of users. There are two important problems with per-item confidentiality: one is inference and the other is the danger of holding back information that may be critical in a certain care situation. Inference means that filtered sensitive information can still be assumed given the other information that was not filtered. The simplest form of inference is that even the existence of a test order for an HIV Western Blot test or a T4/T8 lymphocyte count is a strong indication for an existing HIV infection, even if the results are not known. Very often, diagnoses can be inferred from medication, such as Zidovudin for treatment of HIV infections. The problem of hiding individual items becomes especially difficult with current medications, since the continuing administration of the medication must be assured.

To mitigate some of the inference-risk, aggregations of data should assume the confidentiality level of the most confidential action in the aggregation.

7.1.1.14 Act**.repeatNumber :: IVL<INT>** (0..1)

Attribute description:
An interval of integer numbers stating the minimal and maximal number of repetitions of the Act.

Examples: An oral surgeon's advice to a patient after tooth extraction might be: "replace the gauze every hour for 1 to 3 times until bleeding has stopped completely." This translates to repeatNumber with low boundary 1 and high boundary 3.

Discussion: This attribute is a member of the workflow control suite of attributes.

The number of repeats is additionally constrained by time. The act will repeat at least the minimal number of times and at most, the maximal number of times. Repetitions will also terminate when the time exceeds the maximal Act.effectiveTime, whichever comes first.

7.1.1.15 Act**.interruptibleInd :: BL** (0..1)

Attribute description:
An indicator specifying whether Act is interruptible by asynchronous events.

Discussion: This attribute is part of the suite of workflow control attributes. Act events that are currently active can be interrupted in various ways. Interrupting events include: (1) when an explicit abort request is received against the Act (2) when the time allotted to this Act expires (timeout); (3) a "through condition" ceases to hold true for this Act (see ActRelationship.checkpointCode); (4) the Act is a component with the joinCode "kill" and all other components in that same group have terminated (see Act.joinCode); and (5) when a containing Act is being interrupted.

If an Act receives an interrupt and the Act itself is interruptible, but it has currently active component-Acts that are non-interruptible, the Act will be interrupted when all of its currently active non-interruptible component-acts have terminated.

7.1.1.16 Act**.levelCode :: CE** (0..1)

Vocabulary domain: ActContextLevel (CWE)

Attribute description:
Code specifying the level within a hierarchical Act composition structure and the kind of contextual information attached to composite Acts ("containers") and propagated to component Acts within those containers. The levelCode signifies the position within such a containment hierarchy and the applicable constraints.

Discussion: Readers should be aware that **this attribute may be declared "obsolescent"** in the next normative release of the HL7 RIM. An alternate representation of this concept using a specified hierarchy of Act classCode values is being considered. If the change is adopted, HL7's RIM maintenance procedures state that the levelCode would be declared "obsolescent" in the next RIM release, and then become "obsolete" in the release following that. Users are advised to check with the latest HL7 internal definitions of the RIM, before using this attribute.

The levelCode concepts have been defined to meet specific health record transfer requirements. While these concepts are known to be applicable to some other types of transactions, they are not intended to be a complete closed list. Options exist for other sets of orthogonal levels where required to meet a business purpose (e.g. a multiple patient communication may be subdivided by a super-ordinate level of subject areas).

Examples: The "extract level" and the "folder level" must contain data about a single individual, whereas the "multiple subject level" may contain data about multiple individuals. While "extract" can originate from multiple sources, a "folder" should originate from a single source. The "composition" level usually has a single author.

Constraints: The constraints applicable to a particular level may include differing requirements for participations (e.g. patient, source organization, author or other signatory), relationships to or inclusion of other Acts, documents or use of templates. The constraints pertaining to a level may also specify the permissible levels that may be contained as components of that level. Several nested levels with the same levelCode may be permitted, prohibited (or limited). Instances of the next subordinate level are usually permitted within any level but some levels may be omitted from a model and it may be permissible to skip several layers.

7.1.1.17 Act**.independentInd :: BL** (0..1)

Attribute description:
An indicator specifying whether the Act can be manipulated independently of other Acts or whether manipulation of the Act can only be through a super-ordinate composite Act that has this Act as a component. By default the independentInd should be true.

Examples: An Act definition is sometimes marked with independentInd=false if the business rules would not allow this act to be ordered without ordering the containing act group.

An order may have a component that cannot be aborted independently of the other components.

7.1.1.18 Act**.uncertaintyCode :: CE** (0..1)

Vocabulary domain: ActUncertainty (CNE)

Attribute description:
A code indicating whether the Act statement as a whole, with its subordinate components has been asserted to be uncertain in any way.

Examples: Patient might have had a cholecystectomy procedure in the past (but isn't sure).

Constraints: Uncertainty asserted using this attribute applies to the combined meaning of the Act statement established by all descriptive attributes (e.g., Act.code, Act.effectiveTime, Observation.value, SubstanceAdministration.doseQuantity, etc.), and the meanings of any components.

Discussion: This is not intended for use to replace or compete with uncertainty associated with Observation.value alone or other individual attributes of the class. Such pointed indications of uncertainty should be specified by applying the PPD, UVP or UVN data type extensions to the specific attribute. Particularly if the uncertainty is uncertainty of a quantitative measurement value, this must still be represented by a PPD<PQ> in the value and **not** using the uncertaintyCode. Also, when differential diagnoses are enumerated or weighed for probability, the UVP<CD> or UVN<CD> must be used, not the uncertaintyCode. The use of the uncertaintyCode is appropriate only if the entirety of the Act and its dependent Acts is questioned.

Note that very vague uncertainty may be thought related to negationInd, however, the two concepts are really independent. One may be very uncertain about an event, but that does not mean that one is certain about the negation of the event.

7.1.1.19 Act**.reasonCode :: SET<CE>** (0..*)

Vocabulary domain: ActReason (CWE)

Attribute description:
A code specifying the motivation, cause, or rationale of an Act, when such rationale is not reasonably represented as an ActRelationship of type "has reason" linking to another Act.

Examples: Example reasons that might qualify for being coded in this field might be: "routine requirement", "infectious disease reporting requirement", "on patient request", "required by law".

Discussion

Most reasons for acts can be clearly expressed by linking the new Act to another prior Act using an ActRelationship of type "has reason". This simply states that the prior Act is a reason for the new Act (see ActRelationship). The prior act can then be a specific existing act or a textual explanation. This works for most cases, and the more specific the reason data is, the more should this reason ActRelationship be used instead of the reasonCode.

The reasonCode remains as a place for common reasons that are not related to a prior Act or any other condition expressed in Acts. Indicators that something was required by law or was on the request of a patient etc. may qualify. However, if that piece of legislation, regulation, or the contract or the patient request can be represented as an Act (and they usually can), the reasonCode should not be used.

7.1.1.20 Act**.languageCode :: CE** (0..1)

Vocabulary domain: HumanLanguage (CWE)

Attribute description:
The primary language in which this Act statement is specified, particularly the language of the Act.text.

7.1.1.21 State machine for Act

Link to state diagram for this class.

States of Act:

- **aborted** (sub-state of *normal*): Active service object is exceptionally terminated.
- **active** (sub-state of *normal*): Service object is active.
- **cancelled** (sub-state of *normal*): Service object has been abandoned before activation.
- **completed** (sub-state of *normal*): Service object is completed.
- **held** (sub-state of *normal*): Service object is on hold so it can not be activated before it is released.
- **new** (sub-state of *normal*): Service object is in preparation awaiting to be activated.
- **normal**: Encompasses the expected states of a service object, but excludes "nullified" and "obsolete" which represent unusual terminal states for the life-cycle.
- **nullified**: Service object should not have been instantiated, and is therefore nullified.
- **obsolete**: Service object has been superceded by a new service object.
- **suspended** (sub-state of *normal*): Active service object is temporarily suspended.

State transitions of Act:

- **abort** (from *active* to *aborted*)
- **revise** (from *active* to *active*)
- **complete** (from *active* to *completed*)
- **suspend** (from *active* to *suspended*)
- **reactivate** (from *completed* to *active*)
- **revise** (from *completed* to *completed*)
- **cancel** (from *held* to *cancelled*)
- **revise** (from *held* to *held*)
- **release** (from *held* to *new*)
- **activate** (from *new* to *active*)
- **cancel** (from *new* to *cancelled*)
- **complete** (from *new* to *completed*)
- **hold** (from *new* to *held*)
- **revise** (from *new* to *new*)
- **nullify** (from *normal* to *nullified*)
- **obsolete** (from *normal* to *obsolete*)
- **activate** (from *null* to *active*)
- **complete** (from *null* to *completed*)
- **create** (from *null* to *new*)
- **jump** (from *null* to *normal*)
- **abort** (from *suspended* to *aborted*)
- **resume** (from *suspended* to *active*)
- **complete** (from *suspended* to *completed*)

- **revise** (from *suspended* to *suspended*)

7.1.2 Class: ActRelationship *(in Acts)*

- Attributes of ActRelationship:

typeCode :: CS	priorityNumber :: INT	negationInd :: BL
inversionInd :: BL	pauseQuantity :: PQ	conjunctionCode :: CS
contextControlCode :: CS	checkpointCode :: CS	localVariableName :: ST
contextConductionInd :: BL	splitCode :: CS	seperatableInd :: BL
sequenceNumber :: INT	joinCode :: CS	

- Associations of ActRelationship:

target :: (1..1) Act :: **inboundRelationship** :: (0..*)
source :: (1..1) Act :: **outboundRelationship** :: (0..*)

Description of ActRelationship:

A directed association between a source Act and a target Act. ActRelationship on the same source Act are called the "outbound" act relationships of that Act. ActRelationships on the same target Act are called the "inbound" relationships of that Act. The meaning and purpose of an ActRelationship is specified in the ActRelationship.typeCode attribute.

Examples: 1) An electrolyte observation panel may have sodium, potassium, pH, and bicarbonate observations as components. The composite electrolyte panel would then have 4 outbound ActRelationships of type "has component".

2) The electrolyte panel event has been performed in fulfillment of an observation order. The electrolyte panel event has an outbound ActRelationship of type "fulfills" with the order as target.

3) A Procedure "cholecystectomy" may be performed for the reason of an Observation of "cholelithiasis". The procedure has an outbound ActRelationship of type "has reason" to the cholelithiasis observation.

Discussion: Consider every ActRelationship instance an arrow with a point (headed to the target) and a butt (coming from the source). The functions (sometimes called "roles") that source and target Acts play in that association are defined for each ActRelationship type differently. For instance in a composition relationship, the source is the composite and the target is the component. In a reason-relationship the source is any Act and the target is the reason or indication for the source-Act.

The relationships associated with an Act are considered properties of the source act object. This means that the author of an Act-instance is also considered the author of all of the act relationships that have this Act as their source. There are no exceptions to this rule.

See ActRelationship.typeCode for more overview of the different kinds of ActRelationships.

The ActRelationship class is used to construct action plans and to represent clinical reasoning or judgments about action relationships. Prior actions can be linked as the reasons for more recent actions. Supporting evidence can be linked with current clinical hypotheses. Problem lists and other networks of related judgments about clinical events are represented by the ActRelationship link.

One of the most commonly used ActRelationship types is "has component" to describe the composition and de-composition of Acts. The relationship type allows specifying detail of Acts to varying degrees.

The composition relationship can group actions into "batteries," e.g., LYTES, CHEM12, or CBC, where multiple routine laboratory tests are ordered as a group. Some groupings, such as CHEM12, appear more arbitrary; others, such as blood pressure, seem to naturally consist of systolic and diastolic pressure.

The composition relationships can be arranged in a sequence to form temporal and conditional (non-temporal) action plans (e.g., care plan, critical path, clinical trials protocol, drug treatment protocols). There is a group of attributes in both Act and ActRelationship that we refer to as the "workflow Control suite of attributes", and which allow the detailed specification of executable action plans. These attributes are:

ActRelationship.sequenceNumber arranges the components of an Act as a sequence or as concurrent collections of components, expressing logical branches as well as parallel tasks (tasks carried out at the same time). The ActRelationship attributes splitCode and joinCode control how branches are selected or executable in parallel.

Act.activityTime and ActRelationship.pauseQty allow one to explicitly time an action plan. Act.repeatNumber allows specifying act to repeat (loop).

The ActRelationship type has-precondition allows plan steps to be conditional on the status or outcome of previous actions. The ActRelationhsip.checkpointCode specifies when pre-conditions of acts are tested during the flow of control.

The composition ActRelationship allows these constructs to be organized in multiple layers of nesting to fully support workflow management. This nesting and the workflow control attributes are designed in analogy to a block-structured programming language with support for concurrency (fork, join, interrupts), and without "goto" statements. It is important to note that ALL plans are established through sequencing components (steps) in a composite act (block) as can be depicted in "Nassi-Schneiderman" diagrams (also known as "Chap Charts" or "Structograms"), not by chain-linking acts as in a flowchart diagram.

With the composition relationship, the detail of Acts can be revealed to different levels for different purposes, without the structure of the Act hierarchy needing to be rearranged. This allows supporting multiple viewpoints on the same business processes. For instance, a billing-viewpoint of a laboratory test battery may be as a single billable act. A clinician's view of the same laboratory test battery is as a set of individual observations, where the ordering among the observations is irrelevant. The laboratory's view of this act will be more detailed, including action plan steps that are never reported to the clinician (e.g., centrifugation, decantation, aliquoting, running certain machines etc.). The laboratory's viewpoint warrants a thorough specification of action plans (that can be automated). During this specification, more and more nested sub-activities will be defined. Still the Act is the same, with varying degrees of detail uncovered in the de-composition relationship.

We described the nature of varying detail saying that Acts are "fractal", ever more decomposable, just as the movements of a robotic arm can be decomposed in many fine control steps.

Attributes of ActRelationship:

7.1.2.1 ActRelationship**.typeCode :: CS** (1..1) **Mandatory**

Vocabulary domain: ActRelationshipType (CNE)

Attribute description:
A code specifying the meaning and purpose of every ActRelationship instance. Each of its values implies specific constraints to what kinds of Act objects can be related and in which way.

Discussion: The types of act relationships fall under one of 5 categories:

1.) (De)-composition, with composite (source) and component (target).

2.) Sequel which includes follow-up, fulfillment, instantiation, replacement, transformation, etc. that all have in common that source and target are Acts of essentially the same kind but with variances in mood and other attributes, and where the target exists before the source and the source refers to the target that it links back to.

3.) Pre-condition, trigger, reason, contraindication, with the conditioned Act at the source and the condition or reason at the target.

4.) Post-condition, outcome, goal and risk, with the Act at the source having the outcome or goal at the target.

5.) A host of functional relationships including support, cause, derivation, etc. generalized under the notion of "pertinence".

7.1.2.2 ActRelationship**.inversionInd :: BL** (0..1)

Attribute description:
An indicator specifying that the ActRelationship.typeCode should be interpreted as if the roles of the source and target Acts were reversed. The inversion indicator is used when the meaning of ActRelationship.typeCode must be reversed.

7.1.2.3 ActRelationship**.contextControlCode :: CS** (0..1)

Vocabulary domain: ContextControl (CNE)

Attribute description:
A code that specifies how this ActRelationship contributes to the context of the current Act, and whether it may be propagated to descendent Acts whose association allows such propagation (see ActRelationship.contextConductionInd).

Rationale: In the interest of reducing duplication, humans tend to rely on context when interpreting information. For example, when reading a report taken from a folder containing a patient's medical record, the reader will infer that the report deals with the patient, even if there is no direct reference to the patient on the form. However, other pieces of information such as the author of the folder (the hospital that maintains it) may sometimes apply to the contents of the folder (e.g. a report generated by a doctor at the hospital) and other times not (e.g. a copy of a report from another institution). Humans are quite good at making the necessary inferences about what context should be propagated from an item to something within that item. However, incorrect inferences can occur (perhaps the report in the patient's record deals with a relative). Furthermore, computers have substantially more difficulty making such inferences, even though they can be essential for decision-support systems.

Discussion: This attribute allows the clear specification of whether an association adds to the context associated with a particular item (e.g. adding an additional author) or whether it replaces (overrides) part of the context associated with a particular item (e.g. identifying a sole author, independent of the containing item). It also indicates whether the association applies to only this act (non-propagating), or whether it may apply to derived acts as well (propagating).

This attribute is closely linked with ActRelationship.contextConductionInd which determines whether associations that have been marked as propagating will actually be conducted to a child Act. For example,

an author participation might be marked as propagating, but still not conducted to a hyperlink to an external document.

If no value or default is specified for this attribute (i.e. it is null), no inference can be made about context. Systems must make their own assumptions on the basis of what data is being represented. (For this reason, HL7 committees are encouraged to specify a default or fixed value for this attribute as part of their designs to ensure consistency of interpretation.)

Examples: An observation event has a patient participation marked "additive, propagating" (AP) and has component observation events linked through act relationships that are marked propagating. This means that the patient participation behaves as a patient participation of those component observation events in addition to the parent observation event.

A composite order is created containing a pharmacy order as well as requests for several lab tests. The composite order has participations for patient and author, and an act relationship to a diagnosis, all marked as "additive, propagating". The "component" association between the composite order and the pharmacy order is marked as conductive (contextConductionInd is TRUE). The pharmacy order has an author participation marked as "additive, non-propagating" (AN), and a reason relationship to a diagnosis, marked as "overriding, propagating" (OP). The order further has a relationship to a dispense event, marked as conductive, and an association to a drug protocol marked as non-conductive (contextConductionInd is FALSE). The meaning would be as follows:

The pharmacy order is interpreted as having the patient from the composite order, and having two authors (the one from the composite order, and the one on the pharmacy order itself). The diagnosis for the pharmacy order would **only** be the diagnosis specified on the pharmacy order, not the one specified on the composite order. The dispense event would carry the patient from the composite order and the diagnosis from the pharmacy order, but no author. The drug protocol would not be associated with a patient, diagnosis or author.

7.1.2.4 ActRelationship.contextConductionInd :: BL (0..1)

Attribute description:
If true, associations in the parent act are conducted across the ActRelationship to the child act.

Discussion: Only associations that have been added to the context of an Act and are marked "propagating" will be conducted. (See contextControlCode on ActRelationship and Participation)

The identification of an Act as a parent or child (and therefore the direction context will be conducted) is determined by how the association is traversed when it is serialized. The first act to be encountered is considered to be the parent. Context conducts across the ActRelationship to the second (child) Act.

Refer to ActRelationship.contextControlCode for rationale and examples.

7.1.2.5 ActRelationship.sequenceNumber :: INT (0..1)

Attribute description:
An integer specifying the relative ordering of this relationship among other like-types relationships having the same source Act.

Discussion: This attribute is part of the workflow control suite of attributes. An action plan is a composite Act with component Acts. In a sequential plan, each component has a sequenceNumber that specifies the

ordering of the plan steps. Multiple components with the same sequenceNumber make a branch. Branches can be exclusive (case-switch) or can indicate parallel processes indicated by the splitCode.

7.1.2.6 ActRelationship**.priorityNumber :: INT** (0..1)

Attribute description:
An integer specifying the relative preference for considering this relationship before other like-typed relationships having the same source Act. Relationships with lower priorityNumber values are considered before and above those with higher values.

Examples: For multiple criteria specifies which criteria are considered before others. For components with the same sequence number, specifies which ones are considered before others. Among alternatives or options that are being chosen by humans, the priorityNumber specifies preference.

Discussion: The ordering may be a total ordering in which all priority numbers are unique or a partial ordering, which assigns the same priority to more than one relationship.

7.1.2.7 ActRelationship**.pauseQuantity :: PQ** (0..1)

Attribute description:
A quantity of time that should elapse between when an Act is ready for execution and the actual beginning of the execution.

Discussion: This attribute is part of the workflow control suite of attributes. An action plan is a composite Act with component Acts. In a sequential plan, each component has a sequenceNumber that specifies the ordering of the plan steps. Before each step is executed and has preconditions these conditions are tested and if the test is positive, the Act has clearance for execution. At that time the pauseQuantity timer is started and the Act is executed after the pauseQuantity has elapsed.

7.1.2.8 ActRelationship**.checkpointCode :: CS** (0..1)

Vocabulary domain: ActRelationshipCheckpoint (CNE)

Attribute description:
A code specifying when in the course of an Act a precondition for the Act is evaluated (e.g., before the Act starts for the first time, before every repetition, after each repetition but not before the first, or throughout the entire time of the Act).

Discussion: This attribute is part of the workflow control suite of attributes. An action plan is a composite Act with component Acts. In a sequential plan, each component has a sequenceNumber that specifies the ordering of the plan steps. Before each step is executed, those with preconditions have their conditions tested; where the test is positive, the Act has clearance for execution. The repeatNumber may indicate that an Act may be repeatedly executed. The checkpointCode is specifies when the precondition is checked and is analogous to the various conditional statements and loop constructs in programming languages "while-do" vs. "do-while" or "repeat-until" vs. "loop-exit".

For all checkpointCodes, except "end", preconditions are being checked at the time when the preceding step of the plan has terminated and this step would be next in the sequence established by the sequenceNumber attribute.

When the checkpointCode for a criterion of a repeatable Act is "end", the criterion is tested only at the end of each repetition of that Act. When the condition holds true, the next repetition is ready for execution.

When the checkpointCode is "entry" the criterion is checked at the beginning of each repetition (if any) whereas "beginning" means the criterion is checked only once before the repetition "loop" starts.

The checkpointCode "through" is special in that it requires the condition to hold throughout the execution of the Act, even throughout a single execution. As soon as the condition turns false, the Act should receive an interrupt event (see Act.interruptibleInd) and will eventually terminate.

The checkpointCode "exit" is only used on a special plan step that represents a loop exit step. This allows an action plan to exit due to a condition tested inside the execution of this plan. Such exit criteria are sequenced with the other plan components using the ActRelationship.sequenceNumber.

7.1.2.9 ActRelationship**.splitCode :: CS** (0..1)

Vocabulary domain: ActRelationshipSplit (CNE)

Attribute description:
A code specifying how branches in an action plan are selected among other branches.

Discussion: This attribute is part of the workflow control suite of attributes. An action plan is a composite Act with component Acts. In a sequential plan, each component has a sequenceNumber that specifies the ordering of the plan steps. Branches exist when multiple components have the same sequenceNumber. The splitCode specifies whether a branch is executed exclusively (case-switch) or inclusively, i.e., in parallel with other branches.

In addition to exclusive and inclusive split the splitCode specifies how the pre-condition (also known as "guard conditions" on the branch) are evaluated. A guard condition may be evaluated once when the branching step is entered and if the conditions do not hold at that time, the branch is abandoned. Conversely execution of a branch may wait until the guard condition turns true.

In exclusive wait branches, the first branch whose guard conditions turn true will be executed and all other branches abandoned. In inclusive wait branches some branches may already be executed while other branches still wait for their guard conditions to turn true.

7.1.2.10 ActRelationship**.joinCode :: CS** (0..1)

Vocabulary domain: ActRelationshipJoin (CNE)

Attribute description:
A code specifying how concurrent Acts are resynchronized in a parallel branch construct.

Discussion: This attribute is part of the workflow control suite of attributes. An action plan is a composite Act with component Acts. In a sequential plan, each component has a sequenceNumber that specifies the ordering of the plan steps. Branches exist when multiple components have the same sequenceNumber. Branches are parallel if the splitCode specifies that more than one branch can be executed at the same time. The joinCode then specifies if and how the branches are resynchronized.

The principal re-synchronization actions are (1) the control flow waits for a branch to terminate (wait-branch), (2) the branch that is not yet terminated is aborted (kill-branch), (3) the branch is not re-synchronized at all and continues in parallel (detached branch).

A kill-branch is only executed if there is at least one active wait (or exclusive wait) branch. If there is no other wait branch active, a kill-branch is not started at all (rather than being aborted shortly after it is started).

Since a detached branch is unrelated to all other branches, active detached branches do not protect a kill-branch from being aborted.

7.1.2.11 ActRelationship**.negationInd :: BL** (0..1)

Attribute description:
An indicator that asserts that the meaning of the link is negated.

Examples: If the relationship without negation specifies that Act A has Act B as a component, then the negation indicator specifies that Act A does **not** have Act B as a component. If B is a reason for A, then negation means that B is **not** a reason for A. If B is a pre-condition for A, then negation means that B is **not** a precondition for A.

Discussion: As the examples show, the use of this attribute is quite limited, notably contrast this with the Act.negationInd that actually requires that the described Act not exist, not be done, etc. whereas the ActRelationship.negationInd merely negates this relationship between source and target act, but does not change the meaning of each Act. This is mostly used for clarifying statements.

Note also the difference between negation and the contrary. A contraindication is the contrary of an indication (reason) but not the negation of the reason. The fact that lower back pain is not a reason to prescribe antibiotics doesn't mean that antibiotics are contraindicated with lower back pain.

7.1.2.12 ActRelationship**.conjunctionCode :: CS** (0..1)

Vocabulary domain: RelationshipConjunction (CNE)

Attribute description:
A code specifying the logical conjunction of the criteria among all the condition-links of Acts (e.g., and, or, exclusive-or).

Constraints: All AND criteria must be true. If OR and AND criteria occur together, one criterion out of the OR-group must be true and all AND criteria must be true also. If XOR criteria occur together with OR and AND criteria, exactly one of the XOR criteria must be true, and at least one of the OR criteria and all AND criteria must be true. In other words, the sets of AND, OR, and XOR criteria are in turn combined by a logical AND operator (all AND criteria and at least one OR criterion and exactly one XOR criterion). To overcome this ordering, Act criteria can be nested in any way necessary.

7.1.2.13 ActRelationship**.localVariableName :: ST** (0..1)

Attribute description:
A character string name for the input parameter from which the source Act of this ActRelationship derives some of its attributes. The local variable name is bound in the scope of the Act.derivationExpr with its value being an Act selected based on the input parameter specification.

7.1.2.14 ActRelationship**.seperatableInd :: BL** (0..1)

Attribute description:
This attribute indicates whether or not the source Act is intended to be interpreted independently of the target Act. The indicator cannot prevent an individual or application from separating the Acts, but indicates the author's desire and willingness to attest to the content of the source Act if separated from the target Act. Note that the default for this attribute will typically be "TRUE". Also note that this attribute is orthogonal and unrelated to the RIM's context/inheritance mechanism. If the context of an Act is propagated to nested Acts, it is assumed that those nested Acts are not intended to be interpreted without the propagated context.

7.1.3 Class: Participation *(in Acts)*

- Attributes of Participation:

typeCode :: CS	noteText :: ED	signatureText :: ED
functionCode :: CD	time :: IVL<TS>	performInd :: BL
contextControlCode :: CS	modeCode :: CE	substitutionConditionCode :: CE
sequenceNumber :: INT	awarenessCode :: CE	
negationInd :: BL	signatureCode :: CE	

- Associations of Participation:

  act :: (1..1) Act :: **participation** :: (0..*)
  role :: (1..1) Role :: **participation** :: (0..*)

- Participation generalizes:

  ManagedParticipation

Description of Participation:

An association between an Act and a Role with an Entity playing that Role. Each Entity (in a Role) involved in an Act in a certain way is linked to the act by one Participation-instance. The kind of involvement in the Act is specified by the Participation.typeCode.

Examples: 1) Performers of acts (surgeons, observers, practitioners).

2) Subjects of acts, patient, devices

3) Locations

4) Author, co-signer, witness, informant

5) Addressee, information recipient

Rationale: Participations represent performance while Roles represent competence. Participations specify the actual performance of an Entity in a certain Act, and thus a Participation is scoped by its specific Act. Conversely, Roles specify the competence of an Entity (i.e., how it may principally participate in what kinds of acts) irrespective of any individual Act.

For example, the professional credentials of a person (Role) may be quite different from what a person actually does (Participation). A common example is interns and residents performing anesthesia or surgeries under (more or less) supervision of attending specialists.

An Act can have multiple participations of the same type, which indicates collaborative activities or group involvement. The notion of multiple performing Participations partially overlaps with sub-activities (Act components). Whenever multiple actors are involved in an act, each actor performs a different task (with the extremely rare exception of such symmetrical activities as two people pulling a rope from either end). Thus, the presence of multiple actors could be equally well represented as an act consisting of sub-acts where each act would have only one performing actor

For example, a record of a surgical service may include the actors of type: (a) consenter, (b) primary surgeon, and (c) anesthetist. These three actors really perform different tasks, which can be represented as three related acts: (a) the consent, (b) the surgery proper, and (c) the anesthesia act in parallel to the surgery. If we used the sub-acts, the consenter, surgeon and anesthetist could simply be of actor type "performer."

Thus the more sub-acts we use, the fewer different actor types we need to distinguish; conversely, the fewer sub-acts we use, the more distinct actor types we need.

As a rule of thumb, sub-tasks should be considered instead of multiple actors when each sub-task requires special scheduling, or billing, or if overall responsibilities for the sub-tasks are different. In most cases, however, human resources are scheduled by teams (instead of individuals), billing tends to lump many sub-tasks together into one position, and overall responsibility often rests with one attending physician, chief nurse, or head of department. This model allows both the multi-actor and the multi-act approach to represent the business reality, with a slight bias towards "lumping" minor sub-activities into the overall act.

Attributes of Participation:

7.1.3.1 Participation**.typeCode :: CS** (1..1) **Mandatory**

Vocabulary domain: ParticipationType (CNE)

Attribute description:
A code specifying the kind of Participation or involvement the Entity playing the Role associated with the Participation has with regard to the associated Act.

Constraints: The Participant.typeCode contains only categories that have crisp semantic relevance in the scope of HL7. It is a coded attribute without exceptions and no alternative coding systems allowed.

7.1.3.2 Participation**.functionCode :: CD** (0..1)

Vocabulary domain: ParticipationFunction (CWE)

Attribute description:
An optional code specifying additional detail about the function that the Participation has in the Act, if such detail is not implied by the Participation.typeCode.

Examples: First surgeon, second surgeon (or first assistant surgeon, the one facing the primary surgeon), second assistant (often standing next to the primary surgeon), potentially a third assistant, scrub nurse, circulating nurse, nurse assistant, anesthetist, attending anesthetist, anesthesia nurse, technician who positions the patient, postoperative watch nurse, assistants, midwives, students, etc.

Constraints: This code, if specified at all, **must not** be in conflict with the Participation.typeCode. Automated systems **should not** functionally depend on this code.

No HL7 standard specification may be written to technically depend on the functionCode. If that is deemed necessary, such concepts should be defined in the Participation.typeCode instead.

Discussion: This code can accommodate the huge variety and nuances of functions that participants may perform in the act. The number and kinds of functions applicable depends on the special kind of act. E.g., each operation and method may require a different number of assistant surgeons or nurses.

Since participation functions refer to what people do in an Act, these are really sub-activities that may all occur in parallel. If any more detail needs to be said about these activities other than just who does them, one should consider using component acts instead.

7.1.3.3 Participation**.contextControlCode :: CS** (0..1)

Vocabulary domain: ContextControl (CNE)

Attribute description:
A code that specifies how this Participation contributes to the context of the current Act, and whether it may be propagated to descendent Acts whose association allows such propagation (see ActRelationship.contextConductionInd).

Discussion: Refer to ActRelationship.contextControlCode for rationale, discussion and examples.

7.1.3.4 Participation**.sequenceNumber :: INT** (0..1)

Attribute description:
An integer specifying the relative order of the Participation in relation to other Participations of the same Act.

Rationale: The sequencing might be undertaken for functional reasons or to establish a priority between participations. One example is the sequencing of covered party participations to establish a coordination of benefits sequence in insurance claims.

7.1.3.5 Participation**.negationInd :: BL** (0..1)

Attribute description:
If true, indicates that the specified participation did not, is not or should not occur (depending on mood).

Rationale: This has two primary uses: (1) To indicate that a particular Role did not/should not participate in an Act. (2) To remove a participant from the context being propagated between Acts.

Discussion: A participation with a negationInd set to true is stronger than one with a negationInd of false. In other words, if there is a conflict, the negated participation takes precedence.

Examples: Dr. Smith did not participate; Patient Jones did not sign the consent.

7.1.3.6 Participation**.noteText :: ED** (0..1)

Attribute description:
A textual or multimedia depiction of commentary related to the participation. This note is attributed to this participant only.

7.1.3.7 Participation**.time :: IVL<TS>** (0..1)

Attribute description:
An interval of time specifying the time during which the participant is involved in the act through this Participation.

Rationale: Participation time is needed when the participant's involvement in the act spans only part of the Act's time. Participation time is also used to indicate the time at which certain very common sub-activities happen that are not worth mentioning in full acts.

Examples: 1) The time data was entered into the originating system is the Participation.time of the "data entry" participation.

2) The end of the participation time of an author is the time associated with the signature.

3) The Participation.time of a co-signing participation is also the time of that co-signature.

7.1.3.8 Participation**.modeCode :: CE** (0..1)

Vocabulary domain: ParticipationMode (CWE)

Attribute description:
A code specifying the modality by which the Entity playing the Role is participating in the Act.

Examples: Physically present, over the telephone, written communication.

Rationale: Particularly for author (originator) participants this is used to specify whether the information represented by the act was initially provided verbally, (hand-)written, or electronically.

7.1.3.9 Participation**.awarenessCode :: CE** (0..1)

Vocabulary domain: TargetAwareness (CWE)

Attribute description:
A code specifying the extent to which the Entity playing the participating Role (usually as a target Participation) is aware of the associated Act.

Examples: For diagnostic observations, is the patient, family member or other participant aware of the patient's terminal illness?

Discussion: If the awareness, denial, unconsciousness, etc. is the subject of medical considerations (e.g., part of the problem list), one should use explicit observations in these matters as well, and should not solely rely on this simple attribute in the Participation.

7.1.3.10 Participation**.signatureCode :: CE** (0..1)

Vocabulary domain: ParticipationSignature (CNE)

Attribute description:
A code specifying whether and how the participant has attested his participation through a signature and or whether such a signature is needed.

Examples: A surgical Procedure act object (representing a procedure report) requires a signature of the performing and responsible surgeon, and possibly other participants. (See also: Participation.signatureText.)

7.1.3.11 Participation**.signatureText :: ED** (0..1)

Attribute description:
A textual or multimedia depiction of the signature by which the participant endorses his or her participation in the Act as specified in the Participation.typeCode and that he or she agrees to assume the associated accountability.

Examples: 1) An "author" participant assumes accountability for the truth of the Act statement to the best of his knowledge.

2) An information recipient only attests to the fact that he or she has received the information.

Discussion: The signature can be represented in many different ways either inline or by reference according to the ED data type. Typical cases are:

1) Paper-based signatures: the ED data type may refer to some document or file that can be retrieved through an electronic interface to a hardcopy archive.

2) Electronic signature: this attribute can represent virtually any electronic signature scheme.

3) Digital signature: in particular, this attribute can represent digital signatures, for example, by reference to a signature data block that is constructed in accordance to a digital signature standard, such as XML-DSIG, PKCS#7, PGP, etc.

7.1.3.12 Participation**.performInd :: BL** (0..1)

Attribute description:
Indicates that the resource for this Participation must be reserved before use (i.e. it is controlled by a schedule).

Rationale: This attribute serves a very specific need in the context of resource scheduling. It is not needed in the majority of participation expressions. In most circumstances, it applies to the participation of a particular location or piece of equipment whose use is controlled by a scheduler.

7.1.3.13 Participation**.substitutionConditionCode :: CE** (0..1)

Vocabulary domain: SubstitutionCondition (CWE)

Attribute description:
Indicates the conditions under which a participating item may be substituted with a different one.

7.1.4 Class: Account *(in Acts)*

- Class code: ACCT
- Attributes of Account:

name :: ST	currencyCode :: CE	allowedBalanceQuantity :: IVL<MO>
balanceAmt :: MO	interestRateQuantity :: RTO<MO,PQ>	

- Account is a specialization of: Act

Description of Account:

An Act representing a category of financial transactions that are tracked and reported together with a single balance.

Discussion: This can be used to represent the accumulated total of billable amounts for goods or services received, payments made for goods or services, and debit and credit accounts between which financial transactions flow.

Examples: Patient accounts; Encounter accounts; Cost centers; Accounts receivable

Attributes of Account:

7.1.4.1 Account**.name :: ST** (0..1)

Attribute description:
The descriptive name of the account as carried in the ledger of which the account is a part.

Discussion: This is *not* the identifier for the account. (Use 'id' for account identifiers, name for descriptive/meaningful labels.)

Examples: 'June 2002 Costs'; 'John Smith Receivables'

7.1.4.2 Account**.balanceAmt :: MO** (0..1)

Attribute description:
The total of the debit and credit transactions that have occurred against the account over its lifetime.

Discussion: The balance of an account will generally be communicated in the currency identified as the account's currencyCode. However, it is allowed to communicate the balance in alternative currencies.

7.1.4.3 Account**.currencyCode :: CE** (0..1)

Vocabulary domain: Currency (CWE)

Attribute description:
Indicates the currency that the account is managed in.

Discussion: Specific amounts might be reported in another currency however this represents the default currency for activity in this account.

7.1.4.4 Account**.interestRateQuantity :: RTO<MO,PQ>** (0..1)

Attribute description:
A ratio that indicates the rate of interest that the account balance is subject to, and the term over which the interest rate compounds.

Discussion: This may represent interest charged (e.g. for loans, overdue accounts, etc.) or credited (investments, etc.) depending on the type of account.

Examples: 0.10/1a (10%/year); 0.0005895/1d (.05895%/day)

Constraints: Unit of the denominator PQ data type must be comparable to seconds. (I.e. the denominator must be measured in time.)

7.1.4.5 Account**.allowedBalanceQuantity :: IVL<MO>** (0..1)

Attribute description:
An interval describing the minimum and maximum allowed balances for an account.

Discussion: These are not necessarily 'hard' limits (i.e. the account may go above or below the specified amounts), however, they represent the 'target' range for the account, and there may be consequences for going outside the specified boundaries. It is not necessary to specify both upper and lower limits (or either) for an account.

Examples: 'stop loss' limits; credit limits

7.1.5 Class: DeviceTask *(in Acts)*

- Class code: CONTREG
- Attributes of DeviceTask:

parameterValue :: LIST<ANY>

- DeviceTask is a specialization of: Act

Description of DeviceTask:

An activity of an automated system.

Discussion: Such activities are invoked either by an outside command or are scheduled and executed spontaneously by the device (e.g., regular calibration or flushing). The command to execute the task has moodCode <= ORD; an executed task (including a task in progress) has moodCode <= EVN, an automatic task on the schedule has moodCode <= APT.

Attributes of DeviceTask:

7.1.5.1 DeviceTask**.parameterValue :: LIST<ANY>** (0..*)

Attribute description:
The parameters of the task submitted to the device upon the issuance of a command (or configuring the schedule of spontaneously executed tasks).

Rationale: Some parameters for tasks are uniquely defined by a specific model of equipment. Most critical arguments of a task (e.g., container to operate on, positioning, timing, etc.) are specified in an HL7 standardized structure, and the parameter list would not be used for those. The parameter list is used only for those parameters that cannot be standardized because they are uniquely defined for a specific model of equipment. NOTE: This means that the semantics and interpretation of a parameterValue can **only** be made with an understanding of the specifications or documentation for the specific device being addressed. This information is not conveyed as part of the message.

Constraints: Parameters are only specified here if they are not included in a separate HL7 defined structure. The parameters are a list of any data values interpreted by the device. The parameters should be typed with an appropriate HL7 data type (e.g., codes for nominal settings, such as flags, REAL and INT for numbers, TS for points in time, PQ for dimensioned quantities, etc.). However, besides this HL7 data typing, the functioning of the parameters is opaque to the HL7 standardization.

7.1.6 Class: DiagnosticImage *(in Acts)*

- Class code: DGIMG
- Attributes of DiagnosticImage:
  subjectOrientationCode :: CE
- DiagnosticImage is a specialization of: Observation

Description of DiagnosticImage:

An observation whose immediate and primary outcome (post-condition) is new data about a subject, in the form of visualized images.

Attributes of DiagnosticImage:

7.1.6.1 DiagnosticImage**.subjectOrientationCode :: CE** (0..1)

Vocabulary domain: ImagingSubjectOrientation (CWE)

Attribute description:
A code specifying qualitatively the spatial relation between an imaged object and the imaging film or detector.

7.1.7 Class: Diet *(in Acts)*

- Class code: DIET
- Attributes of Diet:

  energyQuantity :: PQ carbohydrateQuantity :: PQ

- Diet is a specialization of: Supply

Description of Diet:

A supply act dealing specifically with the feeding or nourishment of a subject.

Discussion: The detail of the diet is given as a description of the Material associated via Participation.typeCode="product". Medically relevant diet types may be communicated in the Diet.code, however, the detail of the food supplied and the various combinations of dishes should be communicated as Material instances.

Examples: Gluten free; Low sodium

Attributes of Diet:

7.1.7.1 Diet**.energyQuantity :: PQ** (0..1)

Attribute description:
The supplied biologic energy (Calories) per day.

Discussion: This physical quantity should be convertible to 1 kcal/d (or 1 kJ/d). Note, avoid the existing confusion between "large Calorie" and a "small calorie." Nutrition labels on food products list "large Calories." It is more appropriate to use the small calorie, which is 1/1000 of a large Calorie. These are clearly distinguished in the HL7 units of measure tables.

7.1.7.2 Diet**.carbohydrateQuantity :: PQ** (0..1)

Attribute description:
The supplied amount of carbohydrates (g) per day.

Discussion: For a diabetes diet one typically restricts the amount of metabolized carbohydrates to a certain amount per day (e.g., 240 g/d). This restriction can be communicated in the carbohydrateQuantity.

7.1.8 Class: FinancialContract *(in Acts)*

- Class code: FCNTRCT
- Attributes of FinancialContract:

  paymentTermsCode :: CE

- FinancialContract is a specialization of: Act

Description of FinancialContract:

A contract whose value is measured in monetary terms.

Examples: Insurance; Purchase agreement

Attributes of FinancialContract:

7.1.8.1 FinancialContract**.paymentTermsCode :: CE** (0..1)

Vocabulary domain: PaymentTerms (CWE)

Attribute description:
Establishes the payment terms for a contractual agreement or obligation.

Examples: "net 30"; "on receipt of invoice"; "upon completion of service"

7.1.9 Class: FinancialTransaction *(in Acts)*

- Class code: XACT
- Attributes of FinancialTransaction:

amt :: MO	creditExchangeRateQuantity :: REAL	debitExchangeRateQuantity :: REAL

- FinancialTransaction is a specialization of: Act

Description of FinancialTransaction:

An Act representing the movement of a monetary amount between two accounts.

Discussion: Financial transactions always occur between two accounts (debit and credit), but there may be circumstances where one or both accounts are implied or inherited from the containing model.

In the "order" mood, this represents a request for a transaction to be initiated.

In the "event" mood, this represents the posting of a transaction to an account.

Examples: Cost of a service; Charge for a service; Payment of an invoice

Attributes of FinancialTransaction:

7.1.9.1 FinancialTransaction**.amt :: MO** (0..1)

Attribute description:
Indicates the monetary amount to be transferred from the debit to the credit account.

Discussion: If the denomination of the amt differs from the denomination of the debit or credit account, then the associated exchange rate should be specified.

7.1.9.2 FinancialTransaction**.creditExchangeRateQuantity :: REAL** (0..1)

Attribute description:
A decimal number indicating the rate of exchange in effect between the currency of the account being credited, and the currency of the transaction net amount.

Examples: For the purchase of services valued in Mexican pesos using U.S. dollars paid from a Canadian dollar account, the credit exchange ratio would be communicated as real number "r" such that "y (USD) * r = x (CAD)".

Rationale: This allows a transaction to be expressed in a currency other than that of the credit and debit accounts. It also allows the credit and debit accounts to be based in different currencies.

7.1.9.3 FinancialTransaction**.debitExchangeRateQuantity :: REAL** (0..1)

Attribute description:
A decimal number indicating the rate of exchange in effect between the currency of the account being debited, and the currency of the transaction net amount.

Examples: For the purchase of services valued in Mexican pesos using U.S. dollars paid from a Canadian dollar account, the debit exchange ratio would be communicated as real number "r" such that "y (USD) * r = x (MXP)".

Rationale: This allows a transaction to be expressed in a currency other than that of the credit and debit accounts. It also allows the credit and debit accounts to be based in different currencies.

7.1.10 Class: InvoiceElement *(in Acts)*

- Class code: INVE
- Attributes of InvoiceElement:

modifierCode :: SET<CE>	unitPriceAmt :: RTO<MO,PQ>	factorNumber :: REAL
unitQuantity :: RTO<PQ,PQ>	netAmt :: MO	pointsNumber :: REAL

- InvoiceElement is a specialization of: Act

Description of InvoiceElement:

An Act representing a statement and justification of an "amount owed".

Discussion: This represents the 'justification' portion of an invoice. It is frequently combined with a financial transaction representing the amount requested to be paid, agreed to be paid, or actually paid.

A recursive relationship can be used to break a single InvoiceElement into constituent elements.

In definition mood, it represents "possible" justification for future billing. In request mood, it is a request to determine the amount owed. In event mood, this class represents the determination of a specific amount owed by a particular Entity.

Attributes of InvoiceElement:

7.1.10.1 InvoiceElement**.modifierCode :: SET<CE>** (0..*)

Vocabulary domain: InvoiceElementModifier (CWE)

Attribute description:
Designates a modifier to the code attribute to provide additional information about the invoice element.

Examples: Isolation allowance; After-hours service

Rationale: This is not pre-coordinated into the CD attribute because the modifier code set may not be specifically designed for use with the Act.code code set. This violates the constraint for using the 'modifier' property that the modifier code set must be defined as part of, or specifically for the base code set.

7.1.10.2 InvoiceElement**.unitQuantity :: RTO<PQ,PQ>** (0..1)

Attribute description:
A description of the number of instances of a product or service that is being billed or charged for.

Examples: 4 hours, 4 mg, 4 boxes, and 15 each of a container of 1000 each, etc.

Discussion: Each InvoiceElement that is being charged or billed is identified by a charge or bill code (InvoiceElement.code). In some situations, this code is a pre-coordinated code set and represents a container (e.g. UPC code for a container of 1000 pills and another UPC code for a container of the same pills but in a container of 100). The UPC code is used in invoicing, but ratios are required to specify that only a portion of the container (e.g. bottle) is being billed or charged. If the InvoiceElement does not reference a container, then the denominator is not specified.

For example, 15 pills in a container size of 1000 pills. In this case, the numerator can be expressed as "15 {pill}" or simply "15" and the denominator can be expressed as "1000 {bottle}" or simply "1000" (see discussion following for rationale of using descriptive text for countable units).

Constraints:

The unit of measure is restricted to a measurable unit such as liters, milligrams and hours. Non-measurable, but countable units such as boxes, packages, visits, pills and containers must not be specified using the unit component of the PQ data type, except as an annotation, marked by {xxx}. Refer to Data Types Part II Unabridged Specification, Appendix A :Unified Code for Units of Measure.

Specification of countable units can be handled with the following techniques:

(1) specify the countable unit in the InvoiceElement.code. That is, a specific InvoiceElement.code would indicate that the item referenced by the act represents a box of 20 items. There would be a different InvoiceElement.code for a box of 40 items, and so on.

For example, if the InvoiceElement.code represents a box of 20 items, and the InvoiceElement.unitQuantity = 2 (no units), then this represents 2 boxes of 20 items for a total of 40 items.

(2) If more detail is required (e.g. to describe the composition, packaging, manufacturer of a product), then use a participation (typeCode = "PRD"), and a combination of role and entity classes to describe the details of the packaging.

7.1.10.3 InvoiceElement**.unitPriceAmt :: RTO<MO,PQ>** (0..1)

Attribute description:
The monetary cost per unit being accounted.

Constraints: In constructing the ratio, the numerator must be of data type MO, and the denominator must be a PQ, specified in the same manner as the unitQuantity attribute.

Examples: $0.20/mg; $250/day; $50

7.1.10.4 InvoiceElement**.netAmt :: MO** (0..1)

Attribute description:
Identifies the total monetary amount for the invoice element, including the sum of any component elements.

Discussion: For leaf-level amounts, this will be the value of the unitQuantity * unitPriceAmt [* factorNumber] [* pointsNumber]. For grouping invoice elements, this will be the sum of the netAmt attributes of all contained InvoiceElements.

7.1.10.5 InvoiceElement**.factorNumber :: REAL** (0..1)

Attribute description:
Represents a multiplier used in determining the overall cost of services delivered and/or goods received.

Examples: this could be 10 (Number of Treatments as Units) * $3.00 (Cost per Unit) * 1.5 (Factor) = $45.00 (Amount).

Discussion: This concept is frequently used in Europe to adjust the charge between that used for the public system and that used for private insurers.

The simplest formula for deriving gross amounts is: unitQuantity * unitPriceAmount = netAmt.

The concept of a Factor allows for a discount or surcharge multiplier to be applied to a monetary amount. For example, the formula, with a factor would be: unitQuantity * unitPrice (Cost/Point) * factorNumber = netAmt

See related note on Points. Formula, with Points and Factors becomes: unitQuantity * unitPriceAmt * pointsNumber * factorNumber = netAmt

7.1.10.6 InvoiceElement**.pointsNumber :: REAL** (0..1)

Attribute description:
For charges whose quantity is expressed in 'points', this expresses the weighting (based on difficulty, cost and/or resource intensiveness) associated with the good or service delivered.

Examples: This could be 5 (Number of Treatments as Units) * 3 (Number of Points per treatment as Points)* $20.00 (Cost per Point) = $300.00 (Amount).

Discussion: This is commonly used in systems where services provided are assigned a relative 'cost or difficulty rating', and then a fixed price is assigned to a 'point'. Adjustments to all prices charged by an organization can then be handled by increasing or decreasing the cost per point to reflect changes in inflation, overhead, etc.

The simplest formula for deriving gross amounts is: unitQuantity * unitPriceAmount = netAmt.

The concept of Points allows for assignment of point values for services and/or goods, such that a dollar amount can be assigned to each point. For example, the formula, with points would be: unitQuantity * pointsNumber * unitPriceAmt (Cost/Point) = netAmt.

See related note on Factor. Formula, with Points and Factors becomes: unitQuantity * unitPriceAmt * pointsNumber * factorNumber = netAmt.

7.1.11 Class: ManagedParticipation *(in Acts)*

- Attributes of ManagedParticipation:

  id :: SET<II> statusCode :: SET<CS>

- ManagedParticipation is a specialization of: Participation

Description of ManagedParticipation:

A participation that will be operated on over time and thus whose state and identity must be managed.

Examples: An attending practitioner for an inpatient encounter may change due to leave of absence and it is important to note when this participation will be available.

Rationale: ManagedParticipations are defined as a subclass of Participations to make explicit that not all Participations are stateful. In general, when the sub-activity realized by a Participation is of closer interest and needs to be managed, one SHOULD instead model that sub-activity as an Act component underneath the main Act.

However, in certain environments the view of what these activities really are that the participants perform is not very well recognized and hence modeling those as sub-acts is deemed confusing or burdensome.

Therefore, the ManagedParticipation extends Participation with an identity-attribute and a state-attribute to support these exceptional use cases. ManagedParticipations should be used with utmost caution so as to avoid confusion with Acts and to avoid having to duplicate the act-management infrastructure around participations.

Attributes of ManagedParticipation:

7.1.11.1 ManagedParticipation**.id :: SET<II>** (0..*)

Attribute description:
A unique identifier used to refer to a specific instance of a Participation that may have the same Act and the same Role. (See ManagedParticipation.)

7.1.11.2 ManagedParticipation**.statusCode :: SET<CS>** (0..*)

Vocabulary domain: ManagedParticipationStatus (CNE)

Attribute description:
A code specifying whether the participation instance is pending, active, complete, or cancelled. (See ManagedParticipation.)

7.1.11.3 State machine for ManagedParticipation

Link to state diagram for this class.

States of ManagedParticipation:

- **active** (sub-state of *normal*): The state representing the fact that the Participation is in progress.
- **cancelled** (sub-state of *normal*): The termination state resulting from cancellation of the Participation prior to activation
- **completed** (sub-state of *normal*): The terminal state representing the successful completion of the Participation.
- **normal**: The "typical" state. Excludes "nullified", which represents the termination state of a participation instance that was created in error.
- **nullified**: The state representing the termination of a Participation instance that was created in error.
- **pending** (sub-state of *normal*): The state representing the fact that the Participation has not yet become active.

State transitions of ManagedParticipation:

- **revise** (from *active* to *active*)
- **complete** (from *active* to *completed*)
- **reactivate** (from *completed* to *active*)
- **revise** (from *completed* to *completed*)
- **nullify** (from *normal* to *nullified*)
- **create** (from *null* to *active*)
- **create** (from *null* to *completed*)
- **create** (from *null* to *pending*)
- **activate** (from *pending* to *active*)
- **cancel** (from *pending* to *cancelled*)
- **revise** (from *pending* to *pending*)

7.1.12 Class: Observation *(in Acts)*

- Class code: OBS
- Attributes of Observation:

value :: ANY	methodCode :: SET<CE>
interpretationCode :: SET<CE>	targetSiteCode :: SET<CD>

- Observation generalizes:

DiagnosticImage	PublicHealthCase

- Observation is a specialization of: Act

Description of Observation:

An Act of recognizing and noting information about the subject, and whose immediate and primary outcome (post-condition) is new data about a subject. Observations often involve measurement or other elaborate methods of investigation, but may also be simply assertive statements.

Discussion: Structurally, many observations are name-value-pairs, where the Observation.code (inherited from Act) is the name and the Observation.value is the value of the property. Such a construct is also known as a "variable" (a named feature that can assume a value); hence, the Observation class is always used to hold generic name-value-pairs or variables, even though the variable valuation may not be the result of an elaborate observation method. It may be a simple answer to a question or it may be an assertion or setting of a parameter.

As with all Act statements, Observation statements describe what was done, and in the case of Observations, this includes a description of what was actually observed ("results" or "answers"); and those "results" or "answers" are part of the observation and not split off into other objects.

An observation may consist of component observations each having their own Observation.code and Observation.value. In this case, the composite observation may not have an Observation.value for itself. For instance, a white blood cell count consists of the sub-observations for the counts of the various granulocytes, lymphocytes and other normal or abnormal blood cells (e.g., blasts). The overall white blood cell count Observation itself may therefore not have a value by itself (even though it could have one, e.g., the sum total of white blood cells). Thus, as long as an Act is essentially an Act of recognizing and noting

information about a subject, it is an Observation, regardless of whether it has a simple value by itself or whether it has sub-observations.

Even though observations are professional acts (see Act) and as such are intentional actions, this does not require that every possible outcome of an observation be pondered in advance of it being actually made. For instance, differential white blood cell counts (WBC) rarely show blasts, but if they do, this is part of the WBC observation even though blasts might not be predefined in the structure of a normal WBC.

Diagnoses, findings, symptoms, etc. are also Observations. The Observation.code (or the reference to the Observation definition) specifies the kind of diagnosis (e.g. "chief complaint" or "discharge diagnosis") and the value specifies the diagnosis code or symptom code.

Attributes of Observation:

7.1.12.1 Observation**.value :: ANY** (0..*)

Attribute description:
Information that is assigned or determined by the observation action.

Constraints: The Observation.value, if not otherwise constrained, can be of any data type.

The appropriate data type of the Observation.value varies with the kind of Observation and can generally be described in Observation definitions or in a simple relation that pairs Act.codes to value data types.

The following guidelines govern the choice of the appropriate value data type.

(1) Quantitative measurements use the data type Physical Quantity (PQ) in general. A PQ is essentially a real number with a unit. This is the general preference for all numeric values, subject to a few exceptions listed below.

Numeric values **must not** be communicated as simply a character string (ST).

(2) Titer (e.g., 1:64) and very few other ratios use the data type Ratio (RTO). For titers, the ratio would be a ratio of two integer numbers (e.g., 1:128). Other ratios may relate different quantitative data types, such as a "price" specified in Physical Quantity over Monetary Amount.

Sometimes by local conventions titers are reported as just the denominator (e.g., 32 instead of 1/32). Such conventions are confusing and **should** be converted into correct ratios in HL7 messages.

(3) Index values (a number without unit) uses the Real Number (REAL) data type. When a quantity does not have a proper unit, one can just send the number as a real number. Alternatively one can use a PQ with a dimensionless unit (e.g., 1 or %). An integer number should only be sent when the measurement is by definition an integer, which is an extremely rare case and then is most likely an ordinal (see below).

(4) Ranges (e.g., <3; 12-20) must be expressed as Interval of Physical Quantity (IVL<PQ>) or intervals of other quantity data types.

Sometimes such intervals are used to report the uncertainty of measurement value. For uncertainty there are dedicated data type extensions available.

(5) Ordinals (e.g., +, ++, +++; or I, IIa, IIb, III, IV) use the Coded Ordinal (CO) data type.

(6) Nominal results ("taxons", e.g., organism type). use any of the coded data types (CD, CE) that specify at least a code and a coding system, and optionally original text, translations to other coding systems and sometimes qualifiers.

(7) Imaging results use the Encapsulated Data (ED) type. The encapsulated data type allows one to send an image (e.g., chest X-ray) or a movie (e.g., coronary angiography, cardiac echo) as alternatively inline binary data or as a reference to an external address where the data can be downloaded on demand.

(8) Waveforms can be sent using the Correlated Observation Sequences templates that provide all the data in an HL7 framework. In addition one can use the Encapsulated Data (ED) data type to send waveforms in other than HL7 formats or to refer to waveform data for on-demand download.

(9) The character string data type may exceptionally be used to convey formalized expressions that do not fit in any of the existing data types. However, the string data type **must not** be used if the meaning can be represented by one of the existing data types.

(10) Timestamps **should not** be sent in Observations if there are more appropriate places to send those, e.g., usually as Act.effectiveTime of some act. (E.g., "specimen received in lab" is in the effectiveTime of an Act describing the specimen transport to the lab, not in an Observation.

(11) Sets of values of any data type, enumerated sets as well as intervals, are often used for Observation criteria (event-criterion mood) to specify "normal ranges" or "decision ranges" (for alerts) etc.

(12) For sequences of observations (repeated measurements of the same property during a relatively short time) a Sequence (LIST) data type is used. Refer to the Correlated Observation Sequences specification for more detail.

(13) Uncertainty of values is specified using the Probability and Probability Distribution data type extensions (UVP, PPD). If a statistical sample is reported with absolute frequencies of categories a Bag collection (BAG) can be used efficiently.

7.1.12.2 Observation**.interpretationCode :: SET<CE>** (0..*)

Vocabulary domain: ObservationInterpretation (CWE)

Attribute description:
One or more codes specifying a rough qualitative interpretation of the observation, such as "normal", "abnormal", "below normal", "change up", "resistant", "susceptible", etc.

Discussion: These interpretation codes are sometimes called "abnormal flags", however, the judgment of normalcy is just one of the common rough interpretations, and is often not relevant. For example, the susceptibility interpretations are not about "normalcy", and for any observation of a pathologic condition, it does not make sense to state the normalcy, since pathologic conditions are never considered "normal."

7.1.12.3 Observation**.methodCode :: SET<CE>** (0..*)

Vocabulary domain: ObservationMethod (CWE)

Attribute description:
A code that provides additional detail about the means or technique used to ascertain the observation.

Examples: Blood pressure measurement method: arterial puncture vs. sphygmomanometer (Riva-Rocci), sitting vs. supine position, etc.

Constraints: In all observations the method is already partially specified by the Act.code. In this case, the methodCode **need not** be used at all. The methodCode **may** still be used to identify this method more clearly in addition to what is implied from the Act.code. However, an information consumer system or process **should not** depend on this methodCode information for method detail that is implied by the Act.code.

If the methodCode is used to express method detail that is also implied by the Act.code, the methodCode **must not** be in conflict with the implied method of the Act.code.

Discussion: In all observations the method is already partially specified by simply knowing the kind of observation (observation definition, Act.code) and this implicit information about the method does not need to be specified in Observation.methodCode. For example, many LOINC codes are defined for specific methods as long as the method makes a practical difference in interpretation. Thus, using LOINC, the difference between susceptibility studies using the "minimal inhibitory concentration" (MIC) or the "agar diffusion method" (Kirby-Baur) are specifically assigned different codes. The methodCode therefore is only an additional qualifier to specify what may not be known already from the Act.code.

Also, some variances in methods may be tied to the particular device used. The methodCode should not be used to identify the specific device or test-kit material used in the observation. Such information about devices or test-kits should be associated with the observation as "device" participations.

7.1.12.4 Observation.**targetSiteCode :: SET<CD>** (0..*)

Vocabulary domain: ActSite (CWE)

Attribute description:
A code specifying detail about the anatomical site or system that is the focus of the observation if this information is not already implied by the observation definition or Act.code.

Constraints: The targetSiteCode value, if specified, **must not** conflict with what is implied about the target site or system from the observation definition and the Act.code.

Discussion: Most observation target sites are implied by the observation definition and Act.code, or Observation.value. For example, "heart murmur" always has the heart as target. This attribute is used only when the observation target site needs to be refined, to distinguish right and left etc.

If the subject of the Observation is something other than a human patient or animal, the attribute is used analogously to specify a structural landmark of the thing where the act focuses. For example, if the subject is a lake, the site could be inflow and outflow, etc. If the subject is a lymphatic node, "hilus," "periphery," etc. would still be valid target sites.

7.1.13 Class: PatientEncounter *(in Acts)*

- Class code: ENC
- Attributes of PatientEncounter:

  admissionReferralSourceCode :: CE
  acuityLevelCode :: CE
  preAdmitTestInd :: BL
  specialAccommodationCode :: SET<CE>
  lengthOfStayQuantity :: PQ
  specialCourtesiesCode :: SET<CE>
  dischargeDispositionCode :: CE

- PatientEncounter is a specialization of: Act

Description of PatientEncounter:

An interaction between a patient and care provider(s) for the purpose of providing healthcare-related service(s). Healthcare services include health assessment.

Examples: outpatient visit to multiple departments, home health support (including physical therapy), inpatient hospital stay, emergency room visit, field visit (e.g., traffic accident), office visit, occupational therapy, telephone call.

Attributes of PatientEncounter:

7.1.13.1 PatientEncounter**.admissionReferralSourceCode :: CE** (0..1)

Vocabulary domain: EncounterReferralSource (CWE)

Attribute description:
The source of the referral for a patient encounter.

7.1.13.2 PatientEncounter**.lengthOfStayQuantity :: PQ** (0..1)

Attribute description:
Identifies the total quantity of time when the subject is expected to be or was resident at a facility as part of an encounter.

Discussion: The actual days quantity cannot be simply calculated from the admission and discharge dates because of possible leaves of absence.

7.1.13.3 PatientEncounter**.dischargeDispositionCode :: CE** (0..1)

Vocabulary domain: EncounterDischargeDisposition (CWE)

Attribute description:
A code depicting the actual disposition of the patient at the time of discharge (e.g., discharged to home, expired, against medical advice, etc.).

7.1.13.4 PatientEncounter**.acuityLevelCode :: CE** (0..1)

Vocabulary domain: EncounterAcuity (CWE)

Attribute description:
A code depicting the acuity (complexity of patient care, resource intensiveness of the patient care) of a patient's medical condition upon arrival.

7.1.13.5 PatientEncounter**.preAdmitTestInd :: BL** (0..1)

Attribute description:
An indication that pre-admission tests are required for this patient encounter.

7.1.13.6 PatientEncounter**.specialCourtesiesCode :: SET<CE>** (0..*)

Vocabulary domain: EncounterSpecialCourtesy (CWE)

Attribute description:
A code identifying special courtesies extended to the patient. For example, no courtesies, extended courtesies, professional courtesy, VIP courtesies.

7.1.13.7 PatientEncounter**.specialAccommodationCode :: SET<CE>** (0..*)

Vocabulary domain: SpecialAccommodation (CWE)

Attribute description:
A code indicating the type of special accommodations provided for a patient encounter (e.g., wheelchair, stretcher, interpreter, attendant, seeing eye dog). For encounters in intention moods, this information can be used to identify special arrangements that will need to be made for the incoming patient.

7.1.14 Class: Procedure *(in Acts)*

- Class code: PROC
- Attributes of Procedure:

  methodCode :: SET<CE> approachSiteCode :: SET<CD> targetSiteCode :: SET<CD>

- Procedure is a specialization of: Act

Description of Procedure:

An Act whose immediate and primary outcome (post-condition) is the alteration of the physical condition of the subject.

Examples: Procedures may involve the disruption of some body surface (e.g. an incision in a surgical procedure) conservative procedures such as reduction of a luxated join, including physiotherapy such as chiropractic treatment, massage, balneotherapy, acupuncture, shiatsu, etc. Outside of clinical medicine, procedures may be such things as alteration of environments (e.g. straightening rivers, draining swamps, building dams) or the repair or change of machinery etc.

Discussion: Applied to clinical medicine, procedure is but one among several types of clinical activities such as observation, substance-administrations, and communicative interactions (e.g. teaching, advice, psychotherapy, represented simply as Acts without special attributes). Procedure does not subsume those other activities nor is procedure subsumed by them. Notably Procedure does not comprise all acts of whose intent is intervention or treatment. Whether the bodily alteration is appreciated or intended as beneficial to the subject is likewise irrelevant, what counts is that the act is essentially an alteration of the physical condition of the subject.

The choice between representations for real activities is based on whether the specific properties of procedure are applicable and whether the activity or activity step's necessary post-condition is the physical alteration. For example, taking an x-ray image may sometimes be called "procedure", but it is not a Procedure in the RIM sense for an x-ray image is not done to alter the physical condition of the body.

Many clinical activities combine Acts of Observation and Procedure nature into one composite. For instance, interventional radiology (e.g., catheter directed thrombolysis) does both observing and treating, and most surgical procedures include conscious and documented Observation steps. These clinical activities therefore are best represented by multiple component acts each of the appropriate type.

Attributes of Procedure:

7.1.14.1 Procedure**.methodCode :: SET<CE>** (0..*)

Vocabulary domain: ProcedureMethod (CWE)

Attribute description:
Identifies the means or technique used to perform the procedure.

Discussion: For any Procedure there may be several different methods to achieve by and large the same result, but may be important to know when interpreting a report more thoroughly (e.g., cholecystectomy: open vs. laparoscopic). Method concepts can be "pre-coordinated" in the Act definition. There are many possible methods, which all depend heavily on the particular kind of Procedure, so that defining a vocabulary domain of all methods is difficult. However, a code system might be designed such that it specifies a set of available methods for each defined Procedure concept. Thus, a user ordering a Procedure could select one of several variances of the act by means of the method code. Available method variances may also be defined in a master service catalog for each defined Procedure. In act definition records (Act.moodCode = DEF) the methodCode attribute is a set of all available method codes that a user may select while ordering, or expect while receiving results.

7.1.14.2 Procedure**.approachSiteCode :: SET<CD>** (0..*)

Vocabulary domain: ActSite (CWE)

Attribute description:
The anatomical site or system through which the procedure reaches its target (see targetSiteCode).

Examples:

Nephrectomy can have a trans-abdominal or a primarily retroperitoneal approach

An arteria pulmonalis catheter targets a pulmonary artery but the approach site is typically the vena carotis interna or the vena subclavia, at the neck or the fossa subclavia respectively.

For non-invasive procedures, e.g., acupuncture, the approach site is the punctured area of the skin.

Discussion: If the subject of the Act is something other than a human patient or animal, the attribute is used analogously to specify a structural landmark of the thing where the act focuses.

Some approach sites can also be "pre-coordinated" in the Act definition, so that there is never an option to select different body sites. The same information structure can handle both the pre-coordinated and the post-coordinated approach.

7.1.14.3 Procedure**.targetSiteCode :: SET<CD>** (0..*)

Vocabulary domain: ActSite (CWE)

Attribute description:
The anatomical site or system that is the focus of the procedure.

Examples:

A Nephrectomy's target site is the right or left kidney

An arteria pulmonalis catheter targets a pulmonary artery.

For non-invasive procedures, e.g., acupuncture, the target site is the organ/system that is sought to be influenced (e.g., "the liver").

Discussion: If the subject of the Act is something other than a human patient or animal, the attribute is used analogously to specify a structural landmark of the thing where the act focuses.

Some target sites can also be "pre-coordinated" in the Act definition, so that there is never an option to select different body sites. The same information structure can handle both the pre-coordinated and the post-coordinated approach.

7.1.15 Class: PublicHealthCase *(in Acts)*

- Class code: CASE
- Attributes of PublicHealthCase:
  detectionMethodCode :: CE transmissionModeCode :: CE diseaseImportedCode :: CE
- PublicHealthCase is a specialization of: Observation

Description of PublicHealthCase:

A public health case is an Observation representing a condition or event that has a specific significance for public health. Typically it involves an instance or instances of a reportable infectious disease or other condition. The public health case can include a health-related event concerning a single individual or it may refer to multiple health-related events that are occurrences of the same disease or condition of interest to public health. An outbreak involving multiple individuals may be considered as a type of public health case. A public health case definition (Act.moodCode = "definition") includes the description of the clinical, laboratory, and epidemiologic indicators associated with a disease or condition of interest to public health. There are case definitions for conditions that are reportable, as well as for those that are not. There are also case definitions for outbreaks. A public health case definition is a construct used by public health for the purpose of counting cases, and should not be used as clinical indications for treatment. Examples include AIDS, toxic-shock syndrome, and salmonellosis and their associated indicators that are used to define a case.

Attributes of PublicHealthCase:

7.1.15.1 PublicHealthCase.**detectionMethodCode :: CE** (0..1)

Vocabulary domain: CaseDetectionMethod (CWE)

Attribute description:
Code for the method by which the public health department was made aware of the case. Includes provider report, patient self-referral, laboratory report, case or outbreak investigation, contact investigation, active surveillance, routine physical, prenatal testing, perinatal testing, prison entry screening, occupational disease surveillance, medical record review, etc.

7.1.15.2 PublicHealthCase.**transmissionModeCode :: CE** (0..1)

Vocabulary domain: CaseTransmissionMode (CWE)

Attribute description:
Code for the mechanism by which disease was acquired by the living subject involved in the public health case. Includes sexually transmitted, airborne, bloodborne, vectorborne, foodborne, zoonotic, nosocomial, mechanical, dermal, congenital, environmental exposure, indeterminate.

7.1.15.3 PublicHealthCase**.diseaseImportedCode :: CE** (0..1)

Vocabulary domain: CaseDiseaseImported (CWE)

Attribute description:
Code that indicates whether the disease was likely acquired outside the jurisdiction of observation, and if so, the nature of the inter-jurisdictional relationship. Possible values include not imported, imported from another country, imported from another state, imported from another jurisdiction, and insufficient information to determine.

7.1.16 Class: SubstanceAdministration *(in Acts)*

- Class code: SBADM
- Attributes of SubstanceAdministration:

routeCode :: CE	doseQuantity :: IVL<PQ>	doseCheckQuantity :: SET<RTO>
approachSiteCode :: SET<CD>	rateQuantity :: IVL<PQ>	maxDoseQuantity :: SET<RTO>

- SubstanceAdministration is a specialization of: Act

Description of SubstanceAdministration:

The act of introducing or otherwise applying a substance to the subject.

Discussion: The effect of the substance is typically established on a biochemical basis, however, that is not a requirement. For example, radiotherapy can largely be described in the same way, especially if it is a systemic therapy such as radio-iodine. This class also includes the application of chemical treatments to an area.

Examples: Chemotherapy protocol; Drug prescription; Vaccination record

Attributes of SubstanceAdministration:

7.1.16.1 SubstanceAdministration**.routeCode :: CE** (0..1)

Vocabulary domain: RouteOfAdministration (CWE)

Attribute description:
The method of introducing the therapeutic material into or onto the subject.

Discussion: Route, site of administration (administrationSiteCode) and the device used in administration are closely related. All three (if present) must be closely coordinated and in agreement. In some cases, the coding system used to specify one may pre-coordinate one or more of the others.

When the medication is delivered to an environmental site, or a location, the route code indicates a site on its "body".

Examples: per os (PO), sublingual (SL), rectal (PR), per inhalationem (IH), ophtalmic (OP), nasal (NS), otic (OT), vaginal (VG), intra-dermal (ID), subcutaneous (SC), intra-venous (IV), and intra-cardial (IC)

7.1.16.2 SubstanceAdministration**.approachSiteCode :: SET<CD>** (0..*)

Vocabulary domain: ActSite (CWE)

Attribute description:
The detailed anatomical site where the medication enters or is applied to the subject.

Discussion: This attribute is only needed if the routeCode requires further specification. For example, if the routeCode is "by mouth", no further specification of approach site is needed. If, however, routeCode is intravenous or intra-muscular, the precise site may be specified in this attribute (e.g., right forearm or left deltoid muscle respectively).

Route, site of administration (approachSiteCode) and the device used in administration are closely related. All three (if present) must be closely coordinated and in agreement. In some cases, the coding system used to specify one may pre-coordinate one or more of the others.

7.1.16.3 SubstanceAdministration**.doseQuantity :: IVL<PQ>** (0..1)

Attribute description:
The amount of the therapeutic agent or other substance given at one administration event.

Discussion: The dose may be specified either as a physical quantity of active ingredient (e.g. 200 mg) or as the count of administration-units (e.g., tablets, capsules, "eaches"). Which approach is chosen depends upon the player of the 'consumable' participation (which identifies the drug being administered). If the consumable has a non-countable dosage form (e.g. measured in milligram or litre) then the dose must be expressed in those units. If the consumable has a countable dosage form (tablets, capsules, "eaches"), then the dose must be expressed as a dimensionless count (i.e., with no other unit of measure specified).

The unit of measure is restricted to a measurable unit such as milliliters and milligrams. Non-measurable, but countable units such as tablets and capsules must not be specified using the unit component of the PQ data type, except as an annotation, marked by {xxx}. Refer to Data Types Part II Unabridged Specification, Appendix A :Unified Code for Units of Measure.

7.1.16.4 SubstanceAdministration**.rateQuantity :: IVL<PQ>** (0..1)

Attribute description:
Identifies the speed with which the substance is introduced into the subject. Expressed as a physical (extensive) quantity over elapsed time (e.g., examples are 100 mL/h, 1 g/d, 40 mmol/h, etc.)

Discussion: This is appropriate for continuously divisible dose forms (e.g., liquids, gases). If specified as an interval, the rate should be anywhere in the specified range. This attribute is specified as a extensive physical quantity over elapsed time, i.e., a quantity that could be used for the doseQuantity divided by a duration quantity.

7.1.16.5 SubstanceAdministration**.doseCheckQuantity :: SET<RTO>** (0..*)

Attribute description:
This attribute identifies the expected quantity to be consumed over a period of time. It is used as a verification check on the values specified for other values.

Discussion: This attribute should not generally be used; it is only provided for a special purpose. In some countries, especially Japan, there is a regulatory requirement to note the total daily dose on the prescription

and associated documentation. The purpose of this requirement obviously is to encourage and facilitate reviewing the total dose prescribed to avoid over- (or under-) dosage.

Examples:

With Erythromycin 250 mg 1 tablet 3 times a day one can calculate the total daily dose as "doseCheckQuantity = doseQuantity (1) * Ingredient.quantity (250 mg) * effectiveTime (3 /d) = 750 mg/1d."

For an intravenous example, this term would be "doseCheckQuantity = doseQuantity (100 ml) * Ingredient.quantity (5mg/L) / rateQuantity (1 h) = 0.5 mg/h" which can be calculated on a daily basis as "doseCheckQuantity = 0.5 mg/h * 24 h/d = 12 mg/d."

Rationale: Rather than defining a "total daily dose" attribute as HL7 v2.3 did, we define this general purpose doseCheckQuantity attribute of the Ratio (RTO) data type.

Constraints: invariant(SubstanceAdministration med, RTO max) where med.doseCheckQuantity.contains(max) {max.numerator.compares(med.doseQuantity); max.denominator.compares(1 s);} Numerator must be in units comparable to doseQuantity and denominator must be a measurement of time.

7.1.16.6 SubstanceAdministration**.maxDoseQuantity :: SET<RTO>** (0..*)

Attribute description:
Identifies the maximum total quantity of a therapeutic substance that may be administered to a subject over the period of time.

Discussion: This attribute is particularly useful where the allowed dosage is specified as a range or the timing is variable or PRN (as needed). It provides an overall limit on the quantity that may be administered in a period of time. Multiple occurrences of maxDoseQuantity may be used to indicate different limits over different time periods.

Examples: 500 mg/day; 1200mg/week.

Constraints: invariant(SubstanceAdministration med, RTO max) where med.maxDoseQuantity.contains(max) {max.numerator.compares(med.doseQuantity); max.denominator.compares(1 s);} Numerator must be in units comparable to doseQuantity and denominator must be a measurement of time.

7.1.17 Class: Supply *(in Acts)*

- Class code: SPLY
- Attributes of Supply:

  quantity :: PQ expectedUseTime :: IVL<TS>
- Supply generalizes:

  Diet
- Supply is a specialization of: Act

Description of Supply:

An act that involves provision of a material by one entity to another.

Discussion: The product is associated with the Supply Act via Participation.typeCode="product". With general Supply Acts, the precise identification of the Material (manufacturer, serial numbers, etc.) is important. Most of the detailed information about the Supply should be represented using the Material class. If delivery needs to be scheduled, tracked, and billed separately, one can associate a Transportation Act with the Supply Act. Pharmacy dispense services are represented as Supply Acts, associated with a SubstanceAdministration Act. The SubstanceAdministration class represents the administration of medication, while dispensing is supply.

Examples: Ordering bed sheets; Dispensing of a drug; Issuing medical supplies from storage

Attributes of Supply:

7.1.17.1 Supply**.quantity :: PQ** (0..1)

Attribute description:
The amount that was or is to be supplied (depending on the moodCode)

Discussion: This attribute may be used as an alternative to expectedUseTime or both may be used. If both are specified, then the specified quantity is the amount expected to be consumed within the expectedUseTime.

The unit of measure is restricted to a measured unit such as milliliter and milligram. Non-measured, but countable units such as tablet and capsule must not be specified using the unit component of the PQ data type, except as an annotation, marked by {xxx}. Refer to Data Types Part II Unabridged Specification, Appendix A: Unified Code for Units of Measure. The type of 'countable' information is determined by information in the 'product' entity.

7.1.17.2 Supply**.expectedUseTime :: IVL<TS>** (0..1)

Attribute description:
Identifies the period time over which the supplied product is expected to be used, or the length of time the supply is expected to last.

In some situations, this attribute may be used instead of Supply.quantity to identify the amount supplied by how long it is expected to last, rather than the physical quantity issued. E.g. 90 days supply of medication (based on an ordered dosage), 10 hours of jet fuel, etc. NOTE: When possible, it is always better to specify Supply.quantity, as this tends to be more precise. Supply.expectedUseTime will always be an estimate that can be influenced by external factors.

7.1.18 Class: WorkingList *(in Acts)*

- Class code: LIST
- Attributes of WorkingList:
  ownershipLevelCode :: CE
- WorkingList is a specialization of: Act

Description of WorkingList:

A dynamic list of individual instances of Act which reflect the needs of an individual worker, team of workers, or an organization to view groups of Acts for clinical or administrative reasons.

Discussion: The grouped Acts are related to the WorkingList via an ActRelationship of type 'COMP' (component).

Examples: Problem lists, goal lists, allergy lists, to-do lists, etc.

Attributes of WorkingList:

7.1.18.1 WorkingList**.ownershipLevelCode :: CE** (0..1)

Vocabulary domain: ListOwnershipLevel (CWE)

Attribute description:
Indicates the category of representation for the personnel managing the list, whether person, team or organization.

7.2 Classes in subject area Entities

7.2.1 Class: Entity *(in Entities)*

- Class code: ENT
- Attributes of Entity:

classCode :: CS	quantity :: SET<PQ>	existenceTime :: IVL<TS>
determinerCode :: CS	name :: BAG<EN>	telecom :: BAG<TEL>
id :: SET<II>	desc :: ED	riskCode :: CE
code :: CE	statusCode :: SET<CS>	handlingCode :: CE

- Associations of Entity:

languageCommunication :: (0..*) LanguageCommunication :: **entity** :: (1..1)
playedRole :: (0..*) Role :: **player** :: (0..1)
scopedRole :: (0..*) Role :: **scoper** :: (0..1)

- Entity generalizes:

LivingSubject	Organization
Material	Place

Description of Entity:

A physical thing, group of physical things or an organization capable of participating in Acts, while in a role.

Discussion: An entity is a physical object that has, had or will have existence. The only exception to this is Organization, which while not having a physical presence, fulfills the other characteristics of an Entity. The Entity hierarchy encompasses living subjects (including human beings), organizations, material, and places and their specializations. It does not indicate the roles played, or acts that these entities participate in.

Constraints: It does not include events/acts/actions, or the roles that things can play (e.g. patient, provider).

Attributes of Entity:

7.2.1.1 Entity**.classCode :: CS** (1..1) **Mandatory**

Vocabulary domain: EntityClass (CNE)

Attribute description:
An HL7 defined value representing the class or category that the Entity instance represents.

Examples: Person, Animal, Chemical Substance, Group, Organization

Rationale: Due to the extremely large number of potential values for a code set representing all physical things in the universe, the class code indicates both the subtype branch of the Entity hierarchy used as well as a high level classifier to represent the instance of Entity. This can be used to constrain the eligible value domains for the Entity.code attribute.

7.2.1.2 Entity**.determinerCode :: CS** (1..1) **Mandatory**

Vocabulary domain: EntityDeterminer (CNE)

Attribute description:
An HL7 defined value representing whether the Entity represents a kind-of or a specific instance.

Examples: 1 human being (an instance), 3 syringes (quantified kind) or the population of Indianapolis (kind of group)

Rationale: An Entity may at times represent information concerning a specific instance (the most common), a quantifiable group with common characteristics or a general type of Entity. This code distinguishes these different representations.

7.2.1.3 Entity**.id :: SET<II>** (0..*)

Attribute description:
A unique identifier for the Entity.

Rationale: Successful communication only requires that an entity have a single identifier assigned to it. However, it is recognized that as different systems maintain different databases, there may be different instance identifiers assigned by different systems. Note that an instance identifier is a pure identifier and not a classifier. For Material, serial numbers assigned by specific manufacturers, catalog numbers of specific distributors, or inventory numbers issued by owners, may also be represented by the Role.id, which allows a more clear expression of the fact that such a code is assigned by a specific party associated with that material.

7.2.1.4 Entity**.code :: CE** (0..1)

Vocabulary domain: EntityCode (CWE)

Attribute description:
A value representing the specific kind of Entity the instance represents.

Examples: A medical building, a Doberman Pinscher, a blood collection tube, a tissue biopsy.

Rationale: For each Entity, the value for this attribute is drawn from one of several coding systems depending on the Entity.classCode, such as living subjects (animal and plant taxonomies), chemical substance (e.g., IUPAC code), organizations, insurance company, government agency, hospital, park, lake, syringe, etc. It is possible that Entity.code may be so fine grained that it represents a single instance. An example is the CDC vaccine manufacturer code, modeled as a concept vocabulary, when in fact each concept refers to a single instance.

7.2.1.5 Entity**.quantity :: SET<PQ>** (0..*)

Attribute description:
The number or quantity of the Entity, with appropriate units, congruent with the value of Entity.determinerCode.

Examples: With undetermined kinds, the quantity is but a reference quantity for the specification of the proportion of ingredients or components (e.g. through a has-part, has-ingredient, or has-content Role). For example, a kind of group with 60% females is Person(quantity = 100) has-part Person(quantity = 60; sex = female). Amoxicillin 500 mg per tablet is Material(Tablet, quantity = 1) has-ingredient Material(Amoxicillin, quantity = 500 mg). Glucose 5% (D5W) is Material(D5W, quantity = 1 kg) has-ingredient Material(Glucose, quantity = 50 g).

Material-specific quantity relations are expressed using the fact that the data type of this attribute is a set of physical quantity (SET<PQ>). If more than one quantity value are specified in this set, each element in this set is considered to specify the same amount of the material. For example, for one liter of water one could use the set 1 L, 1 kg, 55.56 mol to specify the volume, mass, and amount of substance for the same amount of water, this is equivalent with specifying the mass density (volumic mass 1 kg/L) and the molar mass (18 g/mol). For Glucose one could specify 180 g, 1 mol according to the molar mass (180 g/mol).

Discussion: When the Entity instance is a portion of a substance, the quantity specifies the amount of that substance comprised by that portion. For an undetermined substance (kind) the quantity serves two purposes at the same time: (a) it provides a means of relations between quantities specific for that substance, and (b) it is a reference quantity for the specification of ingredients or components. In all cases, the quantity is an extensive "amount" kind of quantity (e.g., number, length, volume, mass, surface area, energy, etc.). Note that most relative or fractional quantities are not amounts, in particular, mass fraction, substance concentration, mass ratios, percentages, etc. are not extensive quantities and are prohibited values for this attribute.

Constraints: For Entities with determinerCode = INSTANCE, the quantity is 1. For an Entity with determinerCode = QUANTIFIED_KIND, the quantity is the number of individual members in the group; for an Entity with a determinerCode = KIND, the value is undetermined.

7.2.1.6 Entity**.name :: BAG<EN>** (0..*)

Attribute description:
A non-unique textual identifier or moniker for the Entity.

Examples: Proper names, nicknames, legal names of persons, places or things.

Rationale: Most entities have a commonly used name that can be used to differentiate them from other Entities, but does not provide a unique identifier.

7.2.1.7 Entity**.desc :: ED** (0..1)

Attribute description:
A textual or multimedia depiction of the Entity.

Discussion: The content of the description is not considered part of the functional information communicated between systems. Descriptions are meant to be shown to interested human individuals. All information relevant for automated functions must be communicated using the proper attributes and associated objects.

Rationale: Names and descriptions of entities are typically more meaningful to human viewers than numeric, mnemonic or abbreviated code values. The description allows for additional context about the entity to be conveyed to human viewers without impacting the functional components of the message.

7.2.1.8 Entity**.statusCode :: SET<CS>** (0..*)

Vocabulary domain: EntityStatus (CNE)

Attribute description:
A value representing whether the information associated with the Entity is currently active or inactive for the purpose of participation in acts.

7.2.1.9 Entity**.existenceTime :: IVL<TS>** (0..1)

Attribute description:
An interval of time specifying the period in which the Entity physically existed.

Examples: Birthdate/Deathdate, ManufactureDate/DisposalDate

Constraints: This period may represent past, present or future time periods.

Rationale: Physical entities have specified periods in which they exist. Human beings are born, live their lives and die. Equipment is manufactured, placed in service, retired and salvaged. The relevance of this attribute is in planning, availability and retrospective analysis.

7.2.1.10 Entity**.telecom :: BAG<TEL>** (0..*)

Attribute description:
A telecommunication address for the Entity.

7.2.1.11 Entity**.riskCode :: CE** (0..1)

Vocabulary domain: EntityRisk (CWE)

Attribute description:
A value representing the type of hazard or threat associated with the Entity.

Examples: Petrochemical or organic chemicals are highly flammable agents that pose an increased risk of fire under certain conditions. Entities with either natural or introduced radioactive character pose a risk to those handling them. Entities comprising specimens from diseased individuals pose an increased risk of infection to those handling them. Persons or animals of irascible temperament may prove to be a risk to healthcare personnel.

Rationale: Some entities have characteristics that pose potential increased risk of injury or damage to other Entities. This attribute identifies the type of risk without specifying the level of risk.

7.2.1.12 Entity**.handlingCode :: CE** (0..1)

Vocabulary domain: EntityHandling (CWE)

Attribute description:
A value representing special handling requirements for the Entity.

Examples: Keep at room temperature; Keep frozen below 0 C; Keep in a dry environment; Keep upright, do not turn upside down.

Rationale: This attribute is used to describe special handling required by the Entity to avoid damage to it or other entities.

7.2.1.13 State machine for Entity

Link to state diagram for this class.

States of Entity:

- **active** (sub-state of *normal*): The state representing the fact that the Entity is currently active.
- **inactive** (sub-state of *normal*): The state representing the fact that an entity can no longer be an active participant in events.
- **normal**: The "typical" state. Excludes "nullified", which represents the termination state of an Entity instance that was created in error
- **nullified**: The state representing the termination of an Entity instance that was created in error.

State transitions of Entity:

- **revise** (from *active* to *active*)
- **inactivate** (from *active* to *inactive*)
- **reactivate** (from *inactive* to *active*)
- **revise** (from *inactive* to *inactive*)
- **nullify** (from *normal* to *nullified*)
- **create** (from *null* to *active*)

7.2.2 Class: LanguageCommunication *(in Entities)*

- Attributes of LanguageCommunication:

  languageCode :: CE — proficiencyLevelCode :: CE
  modeCode :: CE — preferenceInd :: BL

- Associations of LanguageCommunication:

  entity :: (1..1) Entity :: **languageCommunication** :: (0..*)

Description of LanguageCommunication:

The language communication capabilities for an Entity.

Examples: A patient who originally came from Mexico may have fluent language capabilities to speak, read and write in Spanish, but only rudimentary capabilities in English. A person from Russia may have the capability to communicate equally well in spoken language in Russian, Armenian or Ukrainian, but prefers to speak in Armenian.

Discussion: While it may seem on the surface that this class would be restricted in usage to only the LivingSubject subtypes, Devices also have the ability to communicate, such as automated telephony devices that transmit patient information to live operators on a triage line or provide automated laboratory results to clinicians.

Rationale: Each Entity with the ability to communicate verbally has differing language and proficiency level. This class specifies the languages with which the entity can communicate, the mode of communication (speak, read, write), the proficiency of that communication, and the Entity's preference.

Attributes of LanguageCommunication:

7.2.2.1 LanguageCommunication**.languageCode :: CE** (0..1)

Vocabulary domain: HumanLanguage (CWE)

Attribute description:
A value representing a language for which the Entity has some level of proficiency for written or spoken communication.

Examples: Spanish, Italian, German, English, American Sign, etc.

Discussion: Communication via spoken or written language is not solely restricted to LivingSubjects. Devices that communicate with persons using human language also must specify in which languages they are capable. Automated voice response systems respond to human language and communicate with other devices or persons using human language.

Rationale: Many individuals and devices have the capability to communicate at varying levels in multiple languages. This code specifies a language capability that the entity wishes to make known.

7.2.2.2 LanguageCommunication**.modeCode :: CE** (0..1)

Vocabulary domain: LanguageAbilityMode (CWE)

Attribute description:
A value representing the method of expression of the language

Examples: expressed spoken, expressed written, expressed signed, received spoken, received written, received signed

7.2.2.3 LanguageCommunication**.proficiencyLevelCode :: CE** (0..1)

Vocabulary domain: LanguageAbilityProficiency (CWE)

Attribute description:
A value representing the level of proficiency in a language.

Examples: excellent, good, fair, poor

7.2.2.4 LanguageCommunication**.preferenceInd :: BL** (0..1)

Attribute description:
An indicator specifying whether or not the language is preferred by the entity for the associated mode.

7.2.3 Class: Container *(in Entities)*

- Class code: CONT
- Attributes of Container:

capacityQuantity :: PQ capTypeCode :: CE bottomDeltaQuantity :: PQ

heightQuantity :: PQ
diameterQuantity :: PQ
separatorTypeCode :: CE
barrierDeltaQuantity :: PQ

- Container is a specialization of: ManufacturedMaterial

Description of Container:

A subtype of ManufacturedMaterial used to hold other Entities for purposes such as transportation or protection of contents from loss or damage.

Rationale: The specifications for this class arose from the collaboration between HL7 and the NCCLS. Many of the attribute definitions are drawn from or reference the NCCLS standard. With amorphic substances (liquids, gases) a container is required. However, the content of a container is always distinguishable and relatively easily separable from the container, unlike the content (ingredient) of a mixture.

Usage: A container is related to a content material through Role.classCode = CONT (content).

Attributes of Container:

7.2.3.1 Container**.capacityQuantity :: PQ** (0..1)

Attribute description:
The functional capacity of the container.

7.2.3.2 Container**.heightQuantity :: PQ** (0..1)

Attribute description:
The height of the container.

7.2.3.3 Container**.diameterQuantity :: PQ** (0..1)

Attribute description:
The outside diameter of the container.

7.2.3.4 Container**.capTypeCode :: CE** (0..1)

Vocabulary domain: ContainerCap (CWE)

Attribute description:
The type of container cap consistent with decapping, piercing or other automated manipulation.

7.2.3.5 Container**.separatorTypeCode :: CE** (0..1)

Vocabulary domain: ContainerSeparator (CWE)

Attribute description:
A material added to a container to facilitate and create a physical separation of specimen components of differing density.

Examples: A gel material added to blood collection tubes that following centrifugation creates a physical barrier between the blood cells and the serum or plasma.

Rationale: The composition or nature of the separator material may have an effect on the analysis. Knowledge of the material aids interpretation of results.

7.2.3.6 Container**.barrierDeltaQuantity :: PQ** (0..1)

Attribute description:
The distance from the Point of Reference to the separator material (barrier) within a container.

Rationale: This distance may be provided by a laboratory automation system to an instrument and/or specimen processing/handling device to facilitate the insertion of a sampling probe into the specimen without touching the separator. See the Point of Reference definition or in NCCLS standard AUTO5 Laboratory Automation: Electromechanical Interfaces.

7.2.3.7 Container**.bottomDeltaQuantity :: PQ** (0..1)

Attribute description:
The distance from the Point of Reference to the outside bottom of the container.

Rationale: Refer to Point of Reference in NCCLS standard AUTO5 Laboratory Automation: Electromechanical Interfaces.

7.2.4 Class: Device *(in Entities)*

- Class code: DEV
- Attributes of Device:

  manufacturerModelName :: SC
  softwareName :: SC
  localRemoteControlStateCode :: CE
  alertLevelCode :: CE
  lastCalibrationTime :: TS

- Device is a specialization of: ManufacturedMaterial

Description of Device:

A subtype of ManufacturedMaterial used in an activity, without being substantially changed through that activity. The kind of device is identified by the code attribute inherited from Entity.

Usage: This includes durable (reusable) medical equipment as well as disposable equipment.

Attributes of Device:

7.2.4.1 Device**.manufacturerModelName :: SC** (0..1)

Vocabulary domain: ManufacturerModelName (CWE)

Attribute description:
The human designated moniker for a device assigned by the manufacturer.

Examples: Perkin Elmer 400 Inductively Coupled Plasma Unit

7.2.4.2 Device**.softwareName :: SC** (0..1)

Vocabulary domain: SoftwareName (CWE)

Attribute description:
The moniker, version and release of the software that operates the device as assigned by the software manufacturer or developer.

Examples: Agilent Technologies Chemstation A.08.xx

7.2.4.3 Device**.localRemoteControlStateCode :: CE** (0..1)

Vocabulary domain: LocalRemoteControlState (CWE)

Attribute description:
A value representing the current state of control associated with the device.

Examples: A device can either work autonomously (localRemoteControlStateCode="Local") or it can be controlled by another system (localRemoteControlStateCode="Remote").

Rationale: The control status of a device must be communicated between devices prior to remote commands being transmitted. If the device is not in "Remote" status then external commands will be ignored.

7.2.4.4 Device**.alertLevelCode :: CE** (0..1)

Vocabulary domain: DeviceAlertLevel (CWE)

Attribute description:
A value representing the current functional activity of an automated device.

Examples: Normal, Warning, Critical

Constraints: The value of the attribute is determined by the device.

7.2.4.5 Device**.lastCalibrationTime :: TS** (0..1)

Attribute description:
The date/time of the last calibration of the device.

Rationale: Devices are required to be recalibrated at specific intervals to ensure they are performing within specifications. The accepted interval between calibrations varies with protocols. Thus for results to be valid, the precise time/date of last calibration is a critical component.

7.2.5 Class: LivingSubject *(in Entities)*

- Class code: LIV
- Attributes of LivingSubject:

administrativeGenderCode :: CE	deceasedTime :: TS	organDonorInd :: BL
birthTime :: TS	multipleBirthInd :: BL	
deceasedInd :: BL	multipleBirthOrderNumber :: INT	

- LivingSubject generalizes:

NonPersonLivingSubject	Person

- LivingSubject is a specialization of: Entity

Description of LivingSubject:

A subtype of Entity representing an organism or complex animal, alive or not.

Examples: A person, dog, microorganism or a plant of any taxonomic group.

Constraints: Instances of this class encompass mammals, birds, fishes, bacteria, parasites, fungi and viruses. Person is a specialization of this class.

Rationale: This class contains "static" or "administrative" attributes of interest to medicine that differentiate living organisms from other Entities.

Attributes of LivingSubject:

7.2.5.1 LivingSubject**.administrativeGenderCode :: CE** (0..1)

Vocabulary domain: AdministrativeGender (CWE)

Attribute description:
A value representing the gender (sex) of a Living subject.

Examples: female, male

Discussion: This attribute does not include terms related to clinical gender. Gender is a complex physiological, genetic and sociological concept that requires multiple observations in order to be comprehensively described. The purpose of this attribute is to provide a high level classification that can additionally be used for the appropriate allocation of inpatient bed assignment.

Constraints: This code is used for administrative purposes.

7.2.5.2 LivingSubject**.birthTime :: TS** (0..1)

Attribute description:
The date and time of a living subject's birth or hatching.

7.2.5.3 LivingSubject**.deceasedInd :: BL** (0..1)

Attribute description:
An indication that the subject is dead.

7.2.5.4 LivingSubject**.deceasedTime :: TS** (0..1)

Attribute description:
The date and time that a living subject's death occurred.

7.2.5.5 LivingSubject**.multipleBirthInd :: BL** (0..1)

Attribute description:
An indication as to whether the living subject is part of a multiple birth.

7.2.5.6 LivingSubject**.multipleBirthOrderNumber :: INT** (0..1)

Attribute description:
The order in which this living subject was born if part of a multiple birth.

7.2.5.7 LivingSubject**.organDonorInd :: BL** (0..1)

Attribute description:
An indication that the living subject is a candidate to serve as an organ donor.

Discussion: This attribute specifies whether an individual living subject has donated or is willing to donate an organ.

7.2.6 Class: ManufacturedMaterial *(in Entities)*

- Class code: MMAT
- Attributes of ManufacturedMaterial:

  lotNumberText :: ST expirationTime :: IVL<TS> stabilityTime :: IVL<TS>

- ManufacturedMaterial generalizes:

  Container Device

- ManufacturedMaterial is a specialization of: Material

Description of ManufacturedMaterial:

A subtype of Material representing an Entity or combination of Entities transformed for a particular purpose by a non-natural or manufacturing process.

Examples: Processed food products, disposable syringes, chemistry analyzer, saline for infusion, etc.

Discussion: This class encompasses containers, devices, software modules and facilities.

Rationale: This class is used to further define the characteristics of Entities that are created out of other Entities. These entities are identified and tracked through associations and mechanisms unique to the class, such as lotName, stabilityTime and expirationTime.

Attributes of ManufacturedMaterial:

7.2.6.1 ManufacturedMaterial**.lotNumberText :: ST** (0..1)

Attribute description:
An alphanumeric string used to identify a particular batch of manufactured material.

Discussion: The lot name is usually printed on the label attached to the container holding the substance and/or on the packaging which houses the container. Note that a lot number is not meant to be a unique identifier, but is meaningful only when the product kind and manufacturer are also identified.

7.2.6.2 ManufacturedMaterial**.expirationTime :: IVL<TS>** (0..1)

Attribute description:
The date and time the manufacturer no longer ensures the safety, quality, and/or proper functioning of the material.

Rationale: There is a need in many situations that the materials used are of a specific quality or potency or functional status. The ending date for this guarantee is specified by the manufacturer. After that date, while the material may still provide the same characteristics, the manufacturer no longer takes responsibility that the product will perform as specified and denies responsibility for failure of the material after that date.

7.2.6.3 ManufacturedMaterial**.stabilityTime :: IVL<TS>** (0..1)

Attribute description:
The time at which the material is considered useable after it is activated.

Examples: After opening a bottle of a liquid. The mixing of two chemicals for an analysis that must be mixed and used within two hours or their activity diminishes.

Discussion: If a kind of material is described (determinerCode = KIND) only the width of that interval can be known, i.e., the duration after opening the reagent container at which the reagent substance is considered useable for its normal testing purpose. For an actual instance of the reagent (e.g., a specific bottle), the stabilityTime.low TS marks the time at which the reagent bottle has been opened (or the reagent was otherwise activated). Together with the typical stability duration, this determines the stabilityTime.high TS beyond which the reagent is no longer considered useable for its normal testing purpose.

7.2.7 Class: Material *(in Entities)*

- Class code: MAT
- Attributes of Material:
  formCode :: CE
- Material generalizes:
  ManufacturedMaterial
- Material is a specialization of: Entity

Description of Material:

A subtype of Entity that is inanimate and locationally independent.

Examples: Pharmaceutical substances (including active vaccines containing retarded virus), disposable supplies, durable equipment, implantable devices, food items (including meat or plant products), waste, traded goods, etc.

Discussion: Manufactured or processed products are considered material, even if they originate as living matter. Materials come in a wide variety of physical forms and can pass through different states (ie. Gas, liquid, solid) while still retaining their physical composition and material characteristics.

Rationale: There are entities that have attributes in addition to the Entity class, yet cannot be classified as either LivingSubject or Place.

Attributes of Material:

7.2.7.1 Material**.formCode :: CE** (0..1)

Vocabulary domain: MaterialForm (CWE)

Attribute description:
A value representing the state (solid, liquid, gas) and nature of the material.

Examples: For therapeutic substances, the dose form, such as tablet, ointment, gel, etc.

7.2.8 Class: NonPersonLivingSubject *(in Entities)*

- Class code: NLIV
- Attributes of NonPersonLivingSubject:
  strainText :: ED genderStatusCode :: CE

- NonPersonLivingSubject is a specialization of: LivingSubject

Description of NonPersonLivingSubject:

A subtype of LivingSubject that includes all living things except the species homo sapiens.

Examples: Cattle, birds, bacteria , plants molds and fungi, etc.

Rationale: Living organisms other than human beings may require additional characterizing information such as genetic strain identification that cannot be conveyed in the Entity.code.

Attributes of NonPersonLivingSubject:

7.2.8.1 NonPersonLivingSubject**.strainText :: ED** (0..1)

Attribute description:
A text string representing a specific genotypic or phenotypic variant of a NonPersonLivingSubject.

Examples: Minnesota5 (swine strain), DXL (poultry strain), RB51 (vaccine strain of Brucella abortus)

Rationale: There is no universal guideline for the naming or cataloging of strains. Many strain designations are created and eliminated over time, while some become established in various industries for a variety of reasons (vaccine production, breeding stock popularity, etc). However, the ability for anyone who cares to designate an organism as a "new" strain, precludes this field from being a coded value. Descriptive text is required to capture these designations.

7.2.8.2 NonPersonLivingSubject**.genderStatusCode :: CE** (0..1)

Vocabulary domain: GenderStatus (CWE)

Attribute description:
A value representing whether the primary reproductive organs of NonPersonLivingSubject are present.

7.2.9 Class: Organization *(in Entities)*

- Class code: ORG
- Attributes of Organization:

  addr :: BAG<AD> standardIndustryClassCode :: CE

- Organization is a specialization of: Entity

Description of Organization:

An Entity representing a formalized group of entities with a common purpose (e.g. administrative, legal, political) and the infrastructure to carry out that purpose.

Examples: Companies and institutions, a government department, an incorporated body that is responsible for administering a facility, an insurance company.

Attributes of Organization:

7.2.9.1 Organization**.addr :: BAG<AD>** (0..*)

Attribute description:
The postal and/or residential address of an organization.

7.2.9.2 Organization**.standardIndustryClassCode :: CE** (0..1)

Vocabulary domain: OrganizationIndustryClass (CWE)

Attribute description:
A value representing the industrial category of an organization entity.

Examples: 11231-Chicken Egg Production, 6211- Offices of Physicians, 621511-Medical Laboratories, 524114-Direct Health and Medical Insurance Carriers

7.2.10 Class: Person *(in Entities)*

- Class code: PSN
- Attributes of Person:

addr :: BAG<AD>	disabilityCode :: SET<CE>	raceCode :: SET<CE>
maritalStatusCode :: CE	livingArrangementCode :: CE	ethnicGroupCode :: SET<CE>
educationLevelCode :: CE	religiousAffiliationCode :: CE	

- Person is a specialization of: LivingSubject

Description of Person:

A subtype of LivingSubject representing a human being.

Constraints: This class can be used to represent either a single individual or a group of individuals based on the value of Entity.determinerCode and Entity.quantity.

Attributes of Person:

7.2.10.1 Person**.addr :: BAG<AD>** (0..*)

Attribute description:
The postal and/or residential address of a Person.

7.2.10.2 Person**.maritalStatusCode :: CE** (0..1)

Vocabulary domain: MaritalStatus (CWE)

Attribute description:
A value representing the domestic partnership status of a person.

Examples: Married, separated, divorced, widowed, common-law marriage.

Rationale: This information is reported on UB FL 16

7.2.10.3 Person**.educationLevelCode :: CE** (0..1)

Vocabulary domain: EducationLevel (CWE)

Attribute description:
The highest level of education a person achieved (e.g. elementary school, high school or secondary school degree complete, college or baccalaureate degree complete).

7.2.10.4 Person**.disabilityCode :: SET<CE>** (0..*)

Vocabulary domain: PersonDisabilityType (CWE)

Attribute description:
A value identifying a person's disability.

Examples: vision impaired, hearing impaired.

7.2.10.5 Person**.livingArrangementCode :: CE** (0..1)

Vocabulary domain: LivingArrangement (CWE)

Attribute description:
A value specifying the housing situation of a person.

Examples: Independent household, institution, nursing home, extended care facility, retirement community, etc.).

Discussion: Used for discharge planning, social service assessment, psychosocial evaluation.

7.2.10.6 Person**.religiousAffiliationCode :: CE** (0..1)

Vocabulary domain: ReligiousAffiliation (CWE)

Attribute description:
The primary religious preference of a person (e.g. Hinduism, Islam, Roman Catholic Church).

7.2.10.7 Person**.raceCode :: SET<CE>** (0..*)

Vocabulary domain: Race (CWE)

Attribute description:
A value representing the race of a person.

7.2.10.8 Person**.ethnicGroupCode :: SET<CE>** (0..*)

Vocabulary domain: Ethnicity (CWE)

Attribute description:
The ethnic group of the person.

7.2.11 Class: Place *(in Entities)*

- Class code: PLC
- Attributes of Place:

mobileInd :: BL
addr :: AD
directionsText :: ED
positionText :: ED
gpsText :: ST

- Place is a specialization of: Entity

Description of Place:

A subtype of Entity representing a bounded physical place or site with its contained structures, if any.

Examples: A field, lake, city, county, state, country, lot (land), building, pipeline, power line, playground, ship, truck.

Constraints: Place may be natural or man-made. The geographic position of a place may or may not be constant.

Discussion: Places may be work facilities (where relevant acts occur), homes (where people live) or offices (where people work). Places may contain sub-places (floor, room, booth, bed). Places may also be sites that are investigated in the context of health care, social work, public health administration (e.g., buildings, picnic grounds, day care centers, prisons, counties, states, and other focuses of epidemiological events).

Attributes of Place:

7.2.11.1 Place**.mobileInd :: BL** (0..1)

Attribute description:
An Indication of whether the facility has the capability to move freely from one location to another.

Examples: Ships, aircraft and ambulances all have the capability to participate in healthcare acts.

7.2.11.2 Place**.addr :: AD** (0..1)

Attribute description:
The physical address of this place.

Constraints: Must be the address that allows the physical location of the place on a map.

7.2.11.3 Place**.directionsText :: ED** (0..1)

Attribute description:
A free text note that carries information related to a site that is useful for entities accessing that site.

Discussion: The attribute could include directions for finding the site when address information is inadequate, GPS information is not available, and/or the entity accessing the site cannot make direct use of the GPS information. It could also include information useful to people visiting the location. E.g., "Last house on the right", "If owner not present, check whereabouts with neighbor down the road".

7.2.11.4 Place**.positionText :: ED** (0..1)

Attribute description:
A set of codes that locates the site within a mapping scheme.

Examples: map coordinates for US Geological Survey maps.

7.2.11.5 Place.**gpsText :: ST** (0..1)

Attribute description:
The global positioning system coordinates of a place.

Discussion: The global positioning system values for this attribute should conform with the USGS Spatial Data Transmission Standards. Among other things this includes the nature of the latitude and longitude readings, the offset error, the projection.

Rationale: In some field conditions, there will be no physical address to identify a place of interest. As all locations on the surface of the earth have unique geographic coordinates, the GPS values allow for precise location information to be captured and transmitted.

7.3 Classes in subject area Roles

7.3.1 Class: Role *(in Roles)*

- Class code: ROL
- Attributes of Role:

classCode :: CS	addr :: BAG<AD>	certificateText :: ED
id :: SET<II>	telecom :: BAG<TEL>	quantity :: RTO
code :: CE	statusCode :: SET<CS>	positionNumber :: LIST<INT>
negationInd :: BL	effectiveTime :: IVL<TS>	

- Associations of Role:

player :: (0..1) Entity :: **playedRole** :: (0..*)
scoper :: (0..1) Entity :: **scopedRole** :: (0..*)
participation :: (0..*) Participation :: **role** :: (1..1)
outboundLink :: (0..*) RoleLink :: **source** :: (1..1)
inboundLink :: (0..*) RoleLink :: **target** :: (1..1)

- Role generalizes:

Access	LicensedEntity
Employee	Patient

Description of Role:

A competency of the Entity playing the Role as identified, defined, guaranteed, or acknowledged by the Entity that scopes the Role.

Discussion: An Entity participates in an Act as in a particular Role. Note that a particular entity in a particular role can participate in an act in many ways. Thus, a Person in the role of a practitioner can participate in a patient encounter as a rounding physician or as an attending physician. The Role defines the competency of the Entity irrespective of any Act, as opposed to Participation, which are limited to the scope of an Act.

Each role is "played" by one Entity, called the "player" and is "scoped" by another Entity, called the "scoper". Thus the Role of "patient" may be played by a person and scoped by the provider organization from which the patient will receive services. Similarly, the employer scopes an "employee" role.

The identifier of the Role identifies the Entity playing the role in that role. This identifier is assigned by the scoping Entity to the player. The scoping Entity need not have issued the identifier, but it may have re-used an existing identifier for the Entity to also identify the Entity in the Role with the scoper.

Most attributes of Role are attributes of the playing entity while in the particular role.

Attributes of Role:

7.3.1.1 Role**.classCode :: CS** (1..1) **Mandatory**

Vocabulary domain: RoleClass (CNE)

Attribute description:
A code specifying the major category of a Role as defined by HL7 vocabulary.

7.3.1.2 Role**.id :: SET<II>** (0..*)

Attribute description:
A unique identifier for the player Entity in this Role.

7.3.1.3 Role**.code :: CE** (0..1)

Vocabulary domain: RoleCode (CWE)

Attribute description:
A code further specifying the kind of Role.

Discussion: The Role.code must conceptually be a proper specialization of Role.classCode. Role.code does not modify Role.classCode. Rather, each is a complete concept or a Role-like relationship between two Entities, but Role.code may be more specific than Role.classCode.

The Role.code may not be coded if only an un-coded name for the type of role is commonly used.

7.3.1.4 Role**.negationInd :: BL** (0..1)

Attribute description:
An indicator specifying that the Role is a competency that is specifically not attributed to the Entity playing the Role.

Examples: 1.) This Person is **not** our Employee

2.) This Mouthwash does **not** have Alcohol as an ingredient.

Constraint

Normally all Roles are considered to be affirmative. (This attribute defaults to FALSE).

7.3.1.5 Role**.addr :: BAG<AD>** (0..*)

Attribute description:
An address for the Entity while in the Role.

7.3.1.6 Role**.telecom :: BAG<TEL>** (0..*)

Attribute description:
A telecommunication address for the Entity while in the Role.

7.3.1.7 Role**.statusCode :: SET<CS>** (0..*)

Vocabulary domain: RoleStatus (CNE)

Attribute description:
A code specifying the state of this Role as defined in the state-transition model.

7.3.1.8 Role**.effectiveTime :: IVL<TS>** (0..1)

Attribute description:
An interval of time specifying the period during which the Role is in effect, if such time limit is applicable and known.

7.3.1.9 Role**.certificateText :: ED** (0..1)

Attribute description:
A textual or multimedia depiction of a certificate issued by the scoping Entity of a Role certifying that this Role is indeed played by the player Entity.

Examples: The certificate can be represented in many different ways, either inline or by reference, according to the ED data type. Typical cases are:

1.) Paper-based certificate: the ED data type may refer to some document or file that can be retrieved through an electronic interface to a hardcopy archive.

2.) Electronic certificate: this attribute can represent virtually any electronic certification scheme, such as, an electronically (including digitally) signed electronic text document.

3.) Digital certificate (public key certificate): in particular, this attribute can represent digital certificates, as an inline data block or by reference to such data. The certificate data block would be constructed in accordance to a digital certificate standard, such as X509, SPKI, PGP, etc.

The certificate subject is the Entity that plays the Role. The certificate issuer is the Entity that scopes the Role.

7.3.1.10 Role**.quantity :: RTO** (0..1)

Attribute description:
A ratio (numerator : denominator) specifying the relative quantities of the Entity playing the Role in the Entity scoping the Role, used for Roles that represent composition relationships between the scoping and playing Entities.

Examples: 1.) This syrup's (scoper) ingredients include 160 mg (numerator) Acetaminophen (player) per tablespoon (denominator).

2.) This herd (scoper) consists of 500 (numerator) cattle (player).

3.) A VAX 6630 computer (scoper) has 3 (numerator) CPUs (player) as parts.

4.) This package (scoper) contains 100 (numerator) pills (player).

Discussion: In composition-relationships (e.g., has-parts, has-ingredient, has-content) the Role.quantity attribute specifies that a numerator amount of the target entity is comprised by a denominator amount of

the source entity of such composition-relationship. For example, if a box (source) has-content 10 eggs (target), the relationship quantity is 10:1; if 0.6 mL contain 75 mg of FeSO4 the ingredient relationship quantity is 75 mg : 0.6 mL. Both numerator and denominator must be amount quantities (extensive quantities, i.e., a count number, mass, volume, amount of substance, amount of energy, etc.).

7.3.1.11 Role**.positionNumber :: LIST<INT>** (0..*)

Attribute description:
An integer specifying the position of the Entity playing the Role with respect to the Entity that scopes the Role.

Discussion: This attribute is primarily used with respect to containment roles. For example, some containers have discrete positions in which content may be located. Depending on the geometry of the container, the position may be referenced as a scalar ordinal number, or as a vector of ordinal numbers (coordinates). Coordinates always begin counting at 1.

Some containers may have customary ways of referring to the positions or no way at all. In the absence of any specific regulation for a specific container type, the rule of thumb is that the coordinate that is changed earlier is positioned first. For an automated blood chemistry analyzer with a square shaped tray, this means that the first coordinate is the one in which direction the tray moves at each step and the second coordinate is the one in which the tray moves only every 10 (or so) steps.

7.3.1.12 State machine for Role

Link to state diagram for this class.

States of Role:

- **active** (sub-state of *normal*): The state representing the fact that the Entity is currently active in the Role.
- **normal**: The "typical" state. Excludes "nullified", which represents the termination of a Role instance that was created in error.
- **nullified**: The state representing the termination of a Role instance that was created in error
- **suspended** (sub-state of *normal*): The state that represents the suspension of the Entity playing the Role. This state is arrived at from the "active" state.
- **terminated** (sub-state of *normal*): The state representing the successful termination of the Role.

State transitions of Role:

- **revise** (from *active* to *active*)
- **suspend** (from *active* to *suspended*)
- **terminate** (from *active* to *terminated*)
- **nullify** (from *normal* to *nullified*)
- **create** (from *null* to *active*)
- **resume** (from *suspended* to *active*)
- **revise** (from *suspended* to *suspended*)
- **terminate** (from *suspended* to *terminated*)
- **reactivate** (from *terminated* to *active*)

- **revise** (from *terminated* to *terminated*)

7.3.2 Class: RoleLink *(in Roles)*

- Attributes of RoleLink:

  typeCode :: CS effectiveTime :: IVL<TS>

- Associations of RoleLink:

  target :: (1..1) Role :: **inboundLink** :: (0..*)
  source :: (1..1) Role :: **outboundLink** :: (0..*)

Description of RoleLink:

A connection between two roles expressing a dependency between those roles.

Examples: 1.) A role of assignment or agency depends on another role of employment, such that when the employment role is terminated, the assignments would be terminated as well. This is the dependency of the assignment role with the employment role, or in other words, the assignment is "part of" the employment.

2.) One role has authority over another (in organizational relationships). For example, an employee of type "manager" may have authority over employees of type "analyst" which would be indicated by a role link for "direct authority".

Discussion: RoleLink specifies the relationships between roles, not between people (or other entities). People (or other Entities) are primarily related by their direct player/scoper relationships around the player's Role and more generally through their interactions (i.e. their participations in acts).

Attributes of RoleLink:

7.3.2.1 RoleLink**.typeCode :: CS** (1..1) **Mandatory**

Vocabulary domain: RoleLinkType (CNE)

Attribute description:
A code specifying the kind of connection represented by this RoleLink, e.g., has-part, has-authority.

7.3.2.2 RoleLink**.effectiveTime :: IVL<TS>** (0..1)

Attribute description:
An interval of time specifying the period during which the connection between Roles is in effect.

7.3.3 Class: Access *(in Roles)*

- Class code: ACCESS
- Attributes of Access:

  approachSiteCode :: CD targetSiteCode :: CD gaugeQuantity :: PQ

- Access is a specialization of: Role

Description of Access:

A role played by a device when the device is used to administer therapeutic agents (medication and vital elements) into the body, or to drain material (e.g., exudates, pus, urine, air, blood) out of the body.

Discussion: In general, Access is a Role of a ManufacturedMaterial or Device, something specifically manufactured or created to serve that purpose, such as a catheter or cannula inserted into a compartment of the body.

Devices in the role of an Access are typically used in intake/outflow observations, and in medication routing instructions. Microbiologic observations on the material itself or on fluids coming out of a drain, are also common.

Rationale: The Access role primarily exists in order to describe material actually deployed as an access, and not so much the fresh material as it comes from the manufacturer. For example, in supply ordering a box of catheters from a distributor, it is not necessary to use the Access role class, since the material attributes will usually suffice to describe and identify the product for the order. But the Access role is used to communicate about the maintenance, intake/outflow, and due replacement of tubes and drains.

Attributes of Access:

7.3.3.1 Access**.approachSiteCode :: CD** (0..1)

Vocabulary domain: ActSite (CWE)

Attribute description:
A coded specification of the anatomic site where the Access (cannula, line or drain) first enters the body and, if applicable, a routing from the first entrance to the target site.

Examples: For example an arteria pulmonalis catheter targets a pulmonary artery but the access approach site is typically the vena carotis interna at the neck, or the vena subclavia at the fossa subclavia.

Constraints: The coding system is the same as for Procedure.approachSiteCode; indeed the Access.approachSiteCode has been copied from the Procedure class into the Access role class. The value of the Access.approachSiteCode should be identical to the value of the Procedure.approachSiteCode of an associated access placement procedure.

Rationale: Since accesses are typically placed for a considerable period of time and since the access is used as a resource of many acts, the access approach site becomes an important identifying attribute of the access itself (as opposed to merely being an attribute of the placement procedure).

7.3.3.2 Access**.targetSiteCode :: CD** (0..1)

Vocabulary domain: ActSite (CWE)

Attribute description:
A coded specification of the site or body compartment into which access is being provided, i.e., the compartment into which material is administered or from which it is collected.

Examples: For example, a pulmonary artery catheter will have the target site "arteria pulmonalis".

Constraints: The coding system is the same as for Procedure.targetSiteCode; indeed the Access.targetSiteCode has been copied from the Procedure class into the Access role class. The value of the Access.targetSiteCode should be identical to the value of the Procedure.targetSiteCode of an associated access placement procedure.

Rationale: Since accesses are typically placed for a considerable period of time and since the access is used as a resource of many acts, the target site becomes an important identifying attribute of the access

itself (as opposed to merely being an attribute of the placement procedure). The target site is important information that determines what kinds of substances may or may not be administered (e.g., special care to avoid medication injections into an arterial access).

7.3.3.3 Access**.gaugeQuantity :: PQ** (0..1)

Attribute description:
A measure for the inner diameter of the Access (e.g. the lumen of the tube).

7.3.4 Class: Employee *(in Roles)*

- Class code: EMP
- Attributes of Employee:

jobCode :: CE	salaryTypeCode :: CE	protectiveEquipmentText :: ED
jobTitleName :: SC	salaryQuantity :: MO	
jobClassCode :: CE	hazardExposureText :: ED	

- Employee is a specialization of: Role

Description of Employee:

A role played by a person who is associated with an organization (the employer, scoper) to receive wages or salary.

Discussion: The purpose of the role is to identify the type of relationship the employee has to the employer rather than the nature of the work actually performed (contrast with AssignedEntity).

Attributes of Employee:

7.3.4.1 Employee**.jobCode :: CE** (0..1)

Vocabulary domain: EmployeeJob (CWE)

Attribute description:
A code specifying the job performed by the employee for the employer. For example, accountant, programmer analyst, patient care associate, staff nurse, etc.

7.3.4.2 Employee**.jobTitleName :: SC** (0..1)

Vocabulary domain: JobTitleName (CWE)

Attribute description:
The title of the job held, for example, Vice President, Senior Technical Analyst.

7.3.4.3 Employee**.jobClassCode :: CE** (0..1)

Vocabulary domain: EmployeeJobClass (CWE)

Attribute description:
A code qualifying the employment in various ways, such as, full-time vs. part-time, etc.

7.3.4.4 Employee**.salaryTypeCode :: CE** (0..1)

Vocabulary domain: EmployeeSalaryType (CWE)

Attribute description:
A code specifying the method used by the employer to compute the employee's salary or wages. For example, hourly, annual, or commission.

7.3.4.5 Employee**.salaryQuantity :: MO** (0..1)

Attribute description:
The amount paid in salary or wages to the employee according to the computation method specified in salaryTypeCode. E.g., if the salaryTypeCode is "hourly" the salaryQuantity specifies the hourly wage.

7.3.4.6 Employee**.hazardExposureText :: ED** (0..1)

Attribute description:
The type of hazards associated with the work performed by the employee for the employer. For example, asbestos, infectious agents.

7.3.4.7 Employee**.protectiveEquipmentText :: ED** (0..1)

Attribute description:
Protective equipment needed for the job performed by the employee for the employer. For example, safety glasses, hardhat.

7.3.5 Class: LicensedEntity *(in Roles)*

- Class code: LIC
- Attributes of LicensedEntity:

  recertificationTime :: TS

- LicensedEntity is a specialization of: Role

Description of LicensedEntity:

A Role of an Entity (player) that is accredited with licenses or qualifications (diplomas) certifying that this Entity may properly perform specific functions.

Examples: 1.) A paramedical training diploma

2.) The certification of equipment

3.) A license to a Person or Organization to provide health services.

Constraints: The scoper is the Organization that issues the credential.

Attributes of LicensedEntity:

7.3.5.1 LicensedEntity**.recertificationTime :: TS** (0..1)

Attribute description:
The date recertification is required.

7.3.6 Class: Patient *(in Roles)*

- Class code: PAT
- Attributes of Patient:

confidentialityCode :: CE veryImportantPersonCode :: CE

- Patient is a specialization of: Role

Description of Patient:

A Role of a LivingSubject (player) as a recipient of health care services from a healthcare provider (scoper).

Attributes of Patient:

7.3.6.1 Patient**.confidentialityCode :: CE** (0..1)

Vocabulary domain: Confidentiality (CWE)

Attribute description:
A code specifying the privacy protection policies in place for this patient.

7.3.6.2 Patient**.veryImportantPersonCode :: CE** (0..1)

Vocabulary domain: PatientImportance (CWE)

Attribute description:
A code specifying the Patient's special status granted by the scoper organization, often resulting in preferred treatment and special considerations. For example, board member, diplomat, etc.

8 Associations in: Health Level Seven Reference Information Model, Release 1

8.1 (0..*) ActRelationship **:: source ::** (1..1) Act **:: outboundRelationship**

8.2 (0..*) ActRelationship **:: target ::** (1..1) Act **:: inboundRelationship**

8.3 (1..1) Entity **:: languageCommunication ::** (0..*) LanguageCommunication **:: entity**

8.4 (0..*) Participation **:: act ::** (1..1) Act **:: participation**

8.5 (0..*) Participation **:: role ::** (1..1) Role **:: participation**

8.6 (0..*) Role **:: player ::** (0..1) Entity **:: playedRole**

The following constraint applies to this association:

Invariant (Role x) { not(x.player.equals(null)) or not(x.scoper.equals(null)) }

8.7 (0..*) Role **:: scoper ::** (0..1) Entity **:: scopedRole**

The following constraint applies to this association:

Invariant (Role x) { not(x.player.equals(null)) or not(x.scoper.equals(null)) }

8.8 (0..*) RoleLink **:: source ::** (1..1) Role **:: outboundLink**

8.9 (0..*) RoleLink **:: target ::** (1..1) Role **:: inboundLink**

9 Normative Vocabulary Contents

9.1 RIM Structural Vocabulary

HL7-defined vocabulary tables provide terms that control the structural attributes of the RIM. These tables define the complete set of specializations for the Act, Role and Entity classes, and specify the types of association represented by the ActRelationship, Participation, and RoleLink classes. Other tables list values for status codes and other controlled concepts in the normative RIM classes.

The tables presented here are organized alphabetically by domain table name or domain name and include a mnemonic code, print name, and definition/description for each coded value. (Abstract domains are not assigned a code). Specifcally:

- The first column contains an integer showing the level of hierarchical containment of the element in that row.
- The second column contains a composite element indented to show the containment hierarchy of the concepts. The opening character of the composite entry indicates whether the concept is: specialized (S) which is both coded and contains child concepts; abstract (A) which does not have a code of its own but does contain child concepts; or a leaf term (Leaf) which is coded but contains no children. The concept name for abstract and specializable domains follows the opening character of the composite element.
- The third column holds the mnemonic code for non-abstract concepts.
- The fourth column holds the print name for non-abstract concepts.
- The final column has the definition of the concept.

The order of the rows is in alphabetic sequence of the domain names at each level of indentation, followed by the leaf terms (in code sequence) at the same level.

This specification contains the vocabulary sets for the following domains:

ActClass
ActMood
ActRelationshipCheckpoint
ActRelationshipJoin
ActRelationshipSplit
ActRelationshipType
ActStatus
ContextControl
EntityClass
EntityDeterminer
EntityStatus
ManagedParticipationStatus
ParticipationType
RelationshipConjunction
RoleClass
RoleLinkType
RoleStatus

ActClass (used in RIM attribute: Act.classCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	**S: ActClassRoot**	ACT	act	(See also the definition of RIM class Act)

An action of interest that has happened, can happen, is happening, is intended to happen, or is requested/demanded to happen. An act is an intentional action in the business domain of HL7. Healthcare (and any profession or business) is constituted of intentional actions. An Act instance is a record of such an intentional action.

Any intentional action can exist in different "moods". Moods describe activities as they progress in the business cycle, from defined, through planned and ordered to complete.

Any instance of an Act assumes one and only one mood and will not change its mood along its life cycle. The moods - definition, intent, order, event - seem to specify a life cycle of an activity. However, the participants in the activity in these different moods are different, as is the data. Therefore, the mood of an Act instance is static. The progression actualization (i.e., the progression from defined, through planned and ordered, to being performed) is called the "business cycle" to distinguish it from the "life cycle" of a single act instance. Related Act instances that form such a "business cycle" are linked through the ActRelationship class.

Examples for acts in health care are: a clinical test, an assessment of health condition (such as problems and diagnoses), the setting of healthcare goals, the performance of treatment services (such as medication, surgery, physical and psychological therapy), assisting, monitoring or attending, training and education services to patients and their next of kin, and notary services (such as advanced directives or living will).

Acts have participants, which can be actors or targets. Examples of actors are nurses, doctors, family members, notary publics, and service organizations -- every person or organization that is capable of independent decisions and can thus is responsible (and liable) for the actions performed.

Target participants in an act may include the patient, the patient's spouse, family, or community, a specimen drawn from the patient or from any object of interest. As patients do play active roles in their own healthcare, the patient can be both an active participant and a target participant at the same time (self-administered or reflexive services).

An act can have multiple active participants and multiple target participants, their specific role being distinguished in the "typeCode" of the respective instance of the Participation class. In particular, an act involving coordination of care may involve two or more active participants -- playing different roles -- who interact on behalf of a patient, family, or aggregate in the role of target participant. For example, a nurse (active participant) calls Meals on Wheels (active participant) on behalf of the patient (target participant).

An act includes the "results", "answers" or informational "procedure products" gained during the act. In this model, "results" do not exist without an act, and every clinical result, including those results gained accidentally, is gleaned via an act. In other moods, such as "definition " or "intent", the results are the possible results, the expected or aimed-for results, or the tested-for results.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	**S: ActClassContract**	CNTRCT	contract	An agreement of obligation between two or more parties that is subject to contractual law and enforcement.
3	**S: ActClassFinancialContract**	FCNTRCT	financial contract	(See also the definition of RIM class FinancialContract) A contract whose value is measured in monetary terms.
4	Leaf	COV	coverage	A health care insurance policy or plan that is contractually binding between two or more parties.
2	**S: ActClassControlAct**	CACT	control act	An act representing a system action such as the change of state of another act or the initiation of a query. All control acts represent trigger events in the HL7 context. ControlActs may occur in different moods.
3	Leaf	ACTN	action	Sender asks addressee to do something depending on the focal Act of the payload. An example is "fulfill this order". Addressee has responsibilities to either reject the message or to act on it in an appropriate way (specified by the specific receiver responsibilities for the interaction).
3	Leaf	INFO	information	Sender sends payload to addressee as information. Addressee does not have responsibilities beyond serving addressee's own interest (i.e., read and memorize if you see fit). This is equivalent to an FYI on a memo.
3	Leaf	STC	state transition control	Sender transmits a status change pertaining to the focal act of the payload. This status of the focal act is the final state of the state transition. This can be either a request or a command, according to the mood of the control act.
2	**S: ActClassObservation**	OBS	observation	(See also the definition of RIM class Observation) Observations are actions performed in order to determine an answer or result value. Observation result values (Observation.value) include specific information about the observed object. The type and constraints of result values depend on the kind of action performed. Clinical documents commonly have 'Subjective' and 'Objective' findings, both of which are kinds of Observations. In addition, clinical documents commonly contain 'Assessments', which are also kinds of Observations. Thus, the establishment of a diagnosis is an Observation.
3	**S: ActClassObservationSeries**	OBSSER	observation series	Container for Correlated Observation Sequences sharing a common frame of reference. All Observations of the same cd must be comparable and relative to the common frame of reference. For example, a 3-channel ECG device records a 12-lead ECG in 4 steps (3 leads at a time). Each of the separate 3-channel recordings would be in their own "OBSCOR". And, all 4 OBSCOR would be contained in one OBSSER because all the times are relative to the same origin (beginning of the recording) and all the ECG signals were from a fixed set of electrodes.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
4	Leaf	OBSCOR	correlated observation sequences	Container for Observation Sequences (Observations whose values are contained in LIST<>'s) having values correlated with each other. Each contained Observation Sequence LIST<> must be the same length. Values in the LIST<>'s are correlated based on index. E.g. the values in position 2 in all the LIST<>'s are correlated. This is analogous to a table where each column is an Observation Sequence with a LIST<> of values, and each row in the table is a correlation between the columns. For example, a 12-lead ECG would contain 13 sequences: one sequence for time, and a sequence for each of the 12 leads.
3	**S: ActClassPublicHealthCase**	CASE	public health case	(See also the definition of RIM class PublicHealthCase) A public health case is an Observation representing a condition or event that has a specific significance for public health. Typically it involves an instance or instances of a reportable infectious disease or other condition. The public health case can include a health-related event concerning a single individual or it may refer to multiple health-related events that are occurrences of the same disease or condition of interest to public health. An outbreak involving multiple individuals may be considered as a type of public health case. A public health case definition (Act.moodCode = "definition") includes the description of the clinical, laboratory, and epidemiologic indicators associated with a disease or condition of interest to public health. There are case definitions for conditions that are reportable, as well as for those that are not. There are also case definitions for outbreaks. A public health case definition is a construct used by public health for the purpose of counting cases, and should not be used as clinical indications for treatment. Examples include AIDS, toxic-shock syndrome, and salmonellosis and their associated indicators that are used to define a case.
4	Leaf	OUTB	outbreak	An outbreak represents a series of public health cases. The date on which an outbreak starts is the earliest date of onset among the cases assigned to the outbreak, and its ending date is the last date of onset among the cases assigned to the outbreak.
3	**A: ActClassROI**			Regions of Interest (ROI) within a subject Act. Primarily used for making secondary observations on a subset of a subject observation. The relationship between a ROI and its referenced Act is specified through an ActRelationship of type "subject" (SUBJ), which must always be present.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
4	Leaf	ROIBND	bounded ROI	A Region of Interest (ROI) specified for a multidimensional observation, such as an Observation Series (OBSSER). The ROI is specified using a set of observation criteria, each delineating the boundary of the region in one of the dimensions in the multidimensional observation. The relationship between a ROI and its referenced Act is specified through an ActRelationship of type subject (SUBJ), which must always be present. Each of the boundary criteria observations is connected with the ROI using ActRelationships of type "has component" (COMP). In each boundary criterion, the Act.code names the dimension and the Observation.value specifies the range of values inside the region. Typically the to be bounded dimension is continuous, and so the Observation.value will be an interval (IVL) data type. The Observation.value need not be specified if the respective dimension is only named but not constrained. For example, an ROI for the QT interval of a certain beat in EKG Lead II would contain 2 boundary criteria, one naming the interval in time (constrained), and the other naming the interval in ECG Lead II (only named, but not constrained).
4	Leaf	ROIOVL	overlay ROI	A Region of Interest (ROI) specified for an image using an overlay shape. Typically used to make reference to specific regions in images, e.g., to specify the location of a radiologic finding in an image or to specify the site of a physical finding by "circling" a region in a schematic picture of a human body. The units of the coordinate values are in pixels. The origin is in the upper left hand corner, with positive X values going to the right and positive Y values going down. The relationship between a ROI and its referenced Act is specified through an ActRelationship of type "subject" (SUBJ), which must always be present.
3	Leaf	ALRT	alert	An observation identifying a potential negative occurrence as a result of an Act or combination of Acts.
3	Leaf	CLNTRL	clinical trial	The set of actions that define an experiment to assess the effectiveness and/or safety of a biopharmaceutical product (food, drug, device, etc.). In definition mood, this set of actions is often embodied in a clinical trial protocol; in event mood, this designates the aggregate act of applying the actions to one or more subjects.
3	Leaf	CNOD	Condition Node	An instance of Observation of a Condition at a point in time that includes any Observations or Procedures associated with that Condition as well as links to previous instances of Condition Node for the same Condition
3	Leaf	COND	Condition	An observable finding or state that persists over time and tends to require intervention or management, and, therefore, distinguished from an Observation made at a point in time; may exist before an Observation of the Condition is made or after interventions to manage the Condition are undertaken. Examples: equipment repair status, device recall status, a health risk, a financial risk, public health risk, pregnancy, health maintenance, chronic illness
3	Leaf	DGIMG	diagnostic image	(See also the definition of RIM class DiagnosticImage) Class for holding attributes unique to diagnostic images.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
3	Leaf	MPROT	monitoring program	An officially or unofficially instituted program to track acts of a particular type or categorization.
3	Leaf	SPCOBS	specimen observation	An observation on a specimen in a laboratory environment that may affect processing, analysis or result interpretation
2	**S: ActClassSupply**	SPLY	supply	(See also the definition of RIM class Supply) Supply orders and deliveries are simple Acts that focus on the delivered product. The product is associated with the Supply Act via Participation.typeCode="product". With general Supply Acts, the precise identification of the Material (manufacturer, serial numbers, etc.) is important. Most of the detailed information about the Supply should be represented using the Material class. If delivery needs to be scheduled, tracked, and billed separately, one can associate a Transportation Act with the Supply Act. Pharmacy dispense services are represented as Supply Acts, associated with a SubstanceAdministration Act. The SubstanceAdministration class represents the administration of medication, while dispensing is supply.
3	Leaf	DIET	diet	(See also the definition of RIM class Diet) Diet services are supply services, with some aspects resembling Medication services: the detail of the diet is given as a description of the Material associated via Participation.typeCode="product". Medically relevant diet types may be communicated in the Diet.code attribute using domain ActDietCode, however, the detail of the food supplied and the various combinations of dishes should be communicated as Material instances.
2	**A: ActDocumentStructureClass**			A structure is a container within a document. Structures have captions which can be coded. Structures can nest, and structures can contain entries.
3	**S: ActClassDocument**	DOC	document	Specialization of Act to add the characteristics unique to document management services.
4	**S: ActClinicalDocument**	DOCCLIN	clinical document	A clinical document is a documentation of clinical observations and services, with the following characteristics: (1) Persistence - A clinical document continues to exist in an unaltered state, for a time period defined by local and regulatory requirements; (2) Stewardship - A clinical document is maintained by a person or organization entrusted with its care; (3) Potential for authentication - A clinical document is an assemblage of information that is intended to be legally authenticated; (4) Wholeness - Authentication of a clinical document applies to the whole and does not apply to portions of the document without the full context of the document; (5) Human readability - A clinical document is human readable."
5	Leaf	CDALVLONE	CDA Level One clinical document	A clinical document that conforms to Level One of the HL7 Clinical Document Architecture (CDA)

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	ACCM	accomodation	An accommodation is a service provided for a Person or other LivingSubject in which a place is provided for the subject to reside for a period of time. Commonly used to track the provision of ward, private and semi-private accommodations for a patient a
2	Leaf	ACCT	account	(See also the definition of RIM class Account) A financial account established to track the net result of financial acts.
2	Leaf	ACSN	accession	A unit of work, a grouper of work items as defined by the system performing that work. Typically some laboratory order fulfillers communicate references to accessions in their communications regarding laboratory orders. Often one or more specimens are related to an accession such that in some environments the accession number is taken as an identifier for a specimen (group).
2	Leaf	ADJUD	financial adjudication	A transformation process where a requested invoice is transformed into an agreed invoice. Represents the adjudication processing of an invoice (claim). Adjudication results can be adjudicated as submitted, with adjustments or refused. Adjudication results comprise 2 components: the adjudication processing results and a restated (or adjudicated) invoice or claim

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	CONS	consent	The Consent class represents informed consents and all similar medico-legal transactions between the patient (or his legal guardian) and the provider. Examples are informed consent for surgical procedures, informed consent for clinical trials, advanced beneficiary notice, against medical advice decline from service, release of information agreement, etc. The details of consents vary. Often an institution has a number of different consent forms for various purposes, including reminding the physician about the topics to mention. Such forms also include patient education material. In electronic medical record communication, consents thus are information-generating acts on their own and need to be managed similar to medical activities. Thus, Consent is modeled as a special class of Act. The "signatures" to the consent document are represented electronically through Participation instances to the consent object. Typically an informed consent has Participation.typeCode of "performer" (the healthcare provider informing the patient, and "consenter", the patient or legal guardian. Some consent may associate a witness or a notary public (e.g., living wills, advanced directives). In consents where a healthcare provider is not required (e.g. living will), the performer may be the patient himself or a notary public. Some consent has a minimum required delay between the consent and the service, so as to allow the patient to rethink his decisions. This minimum delay can be expressed in the act definition by the ActRelationship.pauseQuantity attribute that delays the service until the pause time has elapsed after the consent has been completed.
2	Leaf	CONTREG	container registration	An Act where a container is registered either via an automated sensor, such as a barcode reader, or by manual receipt
2	Leaf	CTTEVENT	clinical trial timepoint event	An identified point during a clinical trial at which one or more actions are scheduled to be performed (definition mood), or are actually performed (event mood). The actions may or may not involve an encounter between the subject and a healthcare professional.
2	Leaf	ENC	encounter	An interaction between a patient and healthcare participant(s) for the purpose of providing patient service(s) or assessing the health status of a patient. For example, outpatient visit to multiple departments, home health support (including physical therapy), inpatient hospital stay, emergency room visit, field visit (e.g., traffic accident), office visit, occupational therapy, telephone call.
2	Leaf	INC	incident	An event that occurred outside of the control of one or more of the parties involved. Includes the concept of an accident.
2	Leaf	INFRM	inform	The act of transmitting information and understanding about a topic to a subject. **Discussion:** This act may be used to request that a patient or provider be informed about an Act, or to indicate that a person was informed about a particular act.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	INVE	invoice element	(See also the definition of RIM class InvoiceElement) Represents concepts related to invoice processing in health care
2	Leaf	LIST	working list	(See also the definition of RIM class WorkingList) WorkingList collects a dynamic list of individual instances of Act via ActRelationship which reflects the need of an individual worker, team of workers, or an organization to manage lists of acts for many different clinical and administrative reasons. Examples of working lists include problem lists, goal lists, allergy lists, and to-do lists.
2	Leaf	PCPR	patient care provision	A PatientCareProvision is the taking on of the responsibility by a performer for the health care of a Patient or group of patients. Discussion: The patient care event can exist without any care actions having taken place. The scope of the care is identified by Act.code. Examples: (1) preferred primary care provision: the primary care physician being the primary performer participation, author being the patient (2) referral from general practitioner to specialist (a PCPR in request mood, where the author participant is the GP, and the primary performer participation is the specialist) (3) a case manager to a patient or group of patients (4) assigning nurses to patients each shift

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	PROC	procedure	(See also the definition of RIM class Procedure) An Act whose immediate and primary outcome (post-condition) is the alteration of the physical condition of the subject. Examples: Procedures may involve the disruption of some body surface (e.g. an incision in a surgical procedure) conservative procedures such as reduction of a luxated join, including physiotherapy such as chiropractic treatment, massage, balneotherapy, acupuncture, shiatsu, etc. Outside of clinical medicine, procedures may be such things as alteration of environments (e.g. straightening rivers, draining swamps, building dams) or the repair or change of machinery etc. Discussion: Applied to clinical medicine, procedure is but one among several types of clinical activities such as observation, substance-administrations, and communicative interactions (e.g. teaching, advice, psychotherapy, represented simply as Acts without special attributes). Procedure does not subsume those other activities nor is procedure subsumed by them. Notably Procedure does not comprise all acts of whose intent is intervention or treatment. Whether the bodily alteration is appreciated or intended as beneficial to the subject is likewise irrelevant, what counts is that the act is essentially an alteration of the physical condition of the subject. The choice between representations for a real activities is based on whether the specific properties of procedure are applicable and whether the activity or activity step's necessary post-condition is the physical alteration. For example, taking an x-ray image may sometimes be called "procedure", but it is not a Procedure in the RIM sense for an x-ray image is not done to alter the physical condition of the body. Many clinical activities combine Acts of Observation and Procedure nature into one composite. For instance, interventional radiology (e.g., catheter directed thrombolysis) does both observing and treating, and most surgical procedures include conscious and documented Observation steps. These clinical activities therefore are best represented by multiple component acts each of the appropriate type.
2	Leaf	REG	registration	Represents the act of maintaining information about an entity or role in a registry. The class is most general, designed to support a variety of registries for persons, patients, practitioners, equipment, etc. If required, specific registry types will be treated as specializations of this class.

Lv	Type, Domain name and/or Mnemonic code	Mnemonic	Print Name	Definition/Description
2	Leaf	SBADM	substance administration	(See also the definition of RIM class SubstanceAdministration) The act of introducing or otherwise applying a substance to the subject. **Discussion:** The effect of the substance is typically established on a biochemical basis, however, that is not a requirement. For example, radiotherapy can largely be described in the same way, especially if it is a systemic therapy such as radio-iodine. This class also includes the application of chemical treatments to an area. **Examples:** Chemotherapy protocol; Drug prescription; Vaccination record
2	Leaf	SPCTRT	specimen treatment	A procedure or treatment performed on a specimen to prepare it for analysis
2	Leaf	TRNS	transportation	Transportation is the moving of a payload (people or material) from a location of origin to a destination location. Thus, any transport service has the three target instances of type payload, origin, and destination, besides the targets that are generally used for any service (i.e., performer, device, etc.)
2	Leaf	XACT	financial transaction	(See also the definition of RIM class FinancialTransaction) A sub-class of Act representing any transaction between two accounts whose value is measured in monetary terms. In the "intent" mood, communicates a request for a transaction to be initiated, or communicates a transfer of value between two accounts. In the "event" mood, communicates the posting of a transaction to an account.

ActMood (used in RIM attribute: Act.moodCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or Mnemonic code	Mnemonic	Print Name	Definition/Description
1	**A: ActMoodCompletionTrack**			These are moods describing activities as they progress in the business cycle, from defined, through planned and ordered to completed.
2	**S: ActMoodIntent**	INT	intent	An intention or plan to perform a service. Historical note: in previous RIM versions, the intent mood was captured as a separate class hierarchy, called Service_intent_or_order.
3	Leaf	APT	appointment	A planned Act for a specific time and place.
3	Leaf	ARQ	appointment request	A request for the booking of an appointment.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
3	Leaf	PRMS	promise	An intent to perform a service that has the strength of a commitment, i.e., other parties may rely on the originator of such promise that said originator will see to it that the promised act will be fulfilled. A promise can be either solicited or unsolicited.
3	Leaf	PRP	proposal	A non-mandated intent to perform an act. Used to record intents that are explicitly not Orders. Professional responsibility for the 'proposal' may or may not be present.
3	Leaf	RQO	request	A request or order for a service is an intent directed from a placer (request author) to a fulfiller (service performer). **Rationale:** The concepts of a "request" and an "order" are viewed as different, because there is an implication of a mandate associated with order. In practice, however, this distinction has no general functional value in the inter-operation of health care computing. "Orders" are commonly refused for a variety of clinical and business reasons, and the notion of a "request" obligates the recipient (the fulfiller) to respond to the sender (the author). Indeed, in many regions, including Australia and Europe, the common term used is "request." Thus, the concept embodies both notions, as there is no useful distinction to be made. If a mandate is to be associated with a request, this will be embodied in the "local" business rules applied to the transactions. Should HL7 desire to provide a distinction between these in the future, the individual concepts could be added as specializations of this concept. The critical distinction here, is the difference between this concept and an "intent", of which it is a specialization. An intent involves decisions by a single party, the author. A request, however, involves decisions by two parties, the author and the fulfiller, with an obligation on the part of the fulfiller to respond to the request indicating that the fulfiller will indeed fulfill the request.
3	Leaf	SLOT	resource slot	Periods of time on a schedule for a resource. Appointments occupy sets of one or more booked slots. A slot that is open for appointments is considered available and a slot that is held back for administrative purposes is considered blocked. A Resource slot that is "tentatively" booked is referred to as reserved.
2	Leaf	DEF	definition	A definition of a service (master). Historical note: in previous RIM versions, the definition mood was captured as a separate class hierarchy, called Master_service.
2	Leaf	EVN	event (occurrence)	A service that actually happens, may be an ongoing service or a documentation of a past service. Historical note: in previous RIM versions, the event mood was captured as a separate class hierarchy, called Patient_service_event, and later Service_event.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	**A: ActMoodPredicate**			Any of the above service moods (e.g., event, intent, or goal) can be turned into a predicate used as a criterion to express conditionals (or queries.) However, currently we allow only criteria on service events.
2	Leaf	EVN.CRT	event criterion	A criterion or condition over service events that must apply for an associated service to be considered.
2	Leaf	GOL	Goal	Expectation to make a specific observation with a desired value at a predefined future time
2	Leaf	OPT	option	An option is an alternative set of property-value bindings. Options specify alternative sets of values, typically used in definitions or orders to describe alternatives. An option can only be used as a group, that is, all assigned values must be used together. Historical note: in HL7 v2.x option existed in the special case for alternative medication routes (RXR segment).

ActRelationshipCheckpoint (used in RIM attribute: ActRelationship.checkpointCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	Leaf	B	beginning	Condition is tested every time before execution of the service (WHILE condition DO service).
1	Leaf	E	end	Condition is tested at the end of a repeated service execution. The service is repeated only if the condition is true (DO service WHILE condition).
1	Leaf	S	entry	Condition is tested once before the service is executed (IF condition THEN service).
1	Leaf	T	through	Condition must be true throughout the execution and the service is interrupted (asynchronously) as soon as the condition turns false (asynchronous WHILE loop). The service must be interruptible.
1	Leaf	X	exit	Condition is a loop checkpoint, i.e. it is a step of an activity plan and, if negative causes the containing loop to exit.

ActRelationshipJoin (used in RIM attribute: ActRelationship.joinCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	Leaf	D	detached	Detach this branch from the other branches so it will not be resynchronized with the other branches.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	Leaf	K	kill	When all other concurrent branches are terminated, interrupt and discontinue this branch.
1	Leaf	W	wait	Wait for this branch to terminate.
1	Leaf	X	exclusive wait	Wait for any one of the branches in the set of exclusive wait branches to terminate, then discontinue all the other exclusive wait branches.

ActRelationshipSplit (used in RIM attribute: ActRelationship.splitCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	Leaf	E1	exclusive try once	The pre-condition associated with the branch is evaluated once and if true the branch may be entered. All other exclusive branches compete with each other and only one will be selected. This implements a COND, IF and CASE conditionals, or "XOR-split." The order in which the branches are considered may be specified in the Service_relationship.priority_nmb.
1	Leaf	EW	exclusive wait	A branch is selected as soon as the pre-condition associated with the branch evaluates to true. If the condition is false, the branch may be entered later, when the condition turns true. All other exclusive branches compete with each other and only one will be selected. Each waiting branch executes in parallel with the default join code wait (see below). The order in which the branches are considered may be specified in the Service_relationship.priority_nmb.
1	Leaf	I1	inclusive try once	A branch is executed if its associated preconditions permit. If associated preconditions do not permit, the branch is dropped. Inclusive branches are not suppressed and do not suppress other branches.
1	Leaf	IW	inclusive wait	A branch is executed as soon as its associated conditions permit. If the condition is false, the branch may be entered later, when the condition turns true. Inclusive branches are not suppressed and do not suppress other branches. Each waiting branch executes in parallel with the default join code wait (see below).

ActRelationshipType (used in RIM attribute: ActRelationship.typeCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	**A: ActRelationshipConditional**			Specifies under what circumstances (target Act) the source-Act may, must, must not or has occurred

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	**S: ActRelationshipReason**	RSON	has reason	The reason or rationale for a service. A reason link is weaker than a trigger, it only suggests that some service may be or might have been a reason for some action, but not that this reason requires/required the action to be taken. Also, as opposed to the trigger, there is no strong timely relation between the reason and the action. **Discussion:** In prior releases, the code "SUGG" (suggests) was expressed as "an inversion of the reason link." That code has been retired in favor of the inversion indicator that is an attribute of ActRelationship.
3	Leaf	MITGT	mitigates	The source act removes or lessens the occurrence or effect of the target act.
2	Leaf	CIND	has contra-indication	A contraindication is just a negation of a reason, i.e. it gives a condition under which the action is not to be done. Both, source and target can be any kind of service, target service is in criterion mood. How the strength of a contraindication is expressed (e.g., relative, absolute) is left as an open issue. The priority_nmb attribute could be used.
2	Leaf	PRCN	has pre-condition	A requirement to be true before a service is performed. The target can be any service in criterion mood. For multiple pre-conditions a conjunction attribute (AND, OR, XOR) is applicable.
2	Leaf	TRIG	has trigger	A pre-condition that if true should result in the source Act being executed. The target is in typically in criterion mood. When reported after the fact (i.e. the criterion has been met) it may be in Event mood. A delay between the trigger and the triggered action can be specified. **Discussion:** This includes the concept of a required act for a service or financial instrument such as an insurance plan or policy. In such cases, the trigger is the occurrence of a specific condition such as coverage limits being exceeded.
1	**S: ActRelationshipHasComponent**	COMP	has component	A collection of sub-services as steps or subtasks performed for the source service. Services may be performed sequentially or concurrently.
2	Leaf	ARR	arrival	The relationship that links to a Transportation Act (target) from another Act (source) indicating that the subject of the source Act entered into the source Act by means of the target Transportation act.
2	Leaf	CTRLV	has control variable	A relationship from an Act to a Control Variable. For example, if a Device makes an Observation, this relates the Observation to its Control Variables documenting the device's settings that influenced the observation.
2	Leaf	DEP	departure	The relationship that links to a Transportation Act (target) from another Act (source) indicating that the subject of the source Act departed from the source Act by means of the target Transportation act.

Lv	Type, Domain name and/or Mnemonic code	Mnemonic	Print Name	Definition/Description
1	**S: ActRelationshipOutcome**	OUTC	has outcome	An observation that should follow or does actually follow as a result or consequence of a condition or action (sometimes called "post-condition"). Target must be an observation as a goal, risk or any criterion. For complex outcomes a conjunction attribute
2	Leaf	GOAL	has goal	A goal that one defines given a patient's health condition. Subsequently planned actions aim to meet that goal. Source is an observation or condition node, target must be an observation in goal mood.
2	Leaf	OBJC	has continuing objective	A desired state that a service action aims to maintain. E.g., keep systolic blood pressure between 90 and 110 mm Hg. Source is an intervention service. Target must be an observation in criterion mood.
2	Leaf	OBJF	has final objective	A desired outcome that a service action aims to meet finally. Source is any service (typically an intervention). Target must be an observation in criterion mood.
2	Leaf	RISK	has risk	A noteworthy undesired outcome of a patient's condition that is either likely enough to become an issue or is less likely but dangerous enough to be addressed.
1	**S: ActRelationshipPertains**	PERT	has pertinent information	This is a very unspecific relationship from one item of clinical information to another. It does not judge about the role the pertinent information plays.
2	**A: ActRelationshipPosts**			Codes that describe the relationship between an Act and a financial instrument such as a financial transaction, account or invoice element.
3	Leaf	CHRG	has charge	A relationship that provides an ability to associate a financial transaction (target) as a charge to a clinical act (source). A clinical act may have a charge associated with the execution or delivery of the service. The financial transaction will define the charge (bill) for delivery or performance of the service. Charges and costs are distinct terms. A charge defines what is charged or billed to another organization or entity within an organization. The cost defines what it costs an organization to perform or deliver a service or product.
3	Leaf	COST	has cost	A relationship that provides an ability to associate a financial transaction (target) as a cost to a clinical act (source). A clinical act may have an inherit cost associated with the execution or delivery of the service. The financial transaction will define the cost of delivery or performance of the service. Charges and costs are distinct terms. A charge defines what is charged or billed to another organization or entity within an organization. The cost defines what it costs an organization to perform or deliver a service or product.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
3	Leaf	CREDIT	has credit	A credit relationship ties a financial transaction to an account. A credit, once applied (posted), may have either a positive or negative effect on the account balance, depending on the type of account. An asset account credit will decrease the account balance. A non-asset account credit will decrease the account balance.
3	Leaf	DEBIT	has debit	A debit relationship ties a financial transaction (target) to an account (source). A debit, once applied (posted), may have either a positive or negative effect on the account balance, depending on the type of account. An asset account debit will increase the account balance. A non-asset account debit will decrease the account balance.
2	**S: has support**	SPRT	has support	Used to indicate that an existing service is suggesting evidence for a new observation. The assumption of support is attributed to the same actor who asserts the observation. Source must be an observation, target may be any service (e.g., to indicate a status post).
3	Leaf	SPRTBND	has bounded support	A specialization of "has support" (SPRT), used to relate a secondary observation to a Region of Interest on a multidimensional observation, if the ROI specifies the true boundaries of the secondary observation as opposed to only marking the approximate area. For example, if the start and end of an ST elevation episode is visible in an EKG, this relation would indicate the ROI bounds the "ST elevation" observation -- the ROI defines the true beginning and ending of the episode. Conversely, if a ROI simply contains ST elevation, but it does not define the bounds (start and end) of the episode, the more general "has support" relation is used. Likewise, if a ROI on an image defines the true bounds of a "1st degree burn", the relation "has bounded support" is used; but if the ROI only points to the approximate area of the burn, the general "has support" relation is used.
2	Leaf	AUTH	authorized by	A relationship in which the target act authorizes or certifies the source act.
2	Leaf	CAUS	is etiology for	An assertion that a new observation was assumed to be the cause for another existing observation. The assumption is attributed to the same actor who asserts the observation. This is stronger and more specific than the support link. For example, a growth of Staphylococcus aureus may be considered the cause of an abscess. The source (cause) is typically an observation, but may be any service, while the target must be an observation.
2	Leaf	COVBY	covered by	A relationship in which the source act is covered by or is under the authority of a target act. A financial instrument such as an Invoice Element is covered by one or more specific instances of an Insurance Policy.
2	Leaf	DRIV	is derived from	A derivation link serves to explicitly associate a derived observation with its input parameters. Both, source and target must be observations, typically numerical observation. E.g., an anion-gap observation can be associated as being derived from given sodium-, (potassium-,) chloride-, and bicarbonate-observations.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	EXPL	has explanation	This is the inversion of support. Used to indicate that a given observation is explained by another observation or condition.
2	Leaf	LIMIT	limited by	A relationship that limits or restricts the source act by the elements of the target act. For example, an authorization may be limited by a financial amount (up to $500). Target Act must be in EVN.CRIT mood.
2	Leaf	MFST	is manifestation of	An assertion that a new observation may be the manifestation of another existing observation or action. This assumption is attributed to the same actor who asserts the manifestation. This is stronger and more specific than an inverted support link. For example, an agitated appearance can be asserted to be the manifestation (effect) of a known hyperthyroxia. This expresses that one might not have realized a symptom if it would not be a common manifestation of a known condition. The target (cause) may be any service, while the source (manifestation) must be an observation.
2	Leaf	NAME	assigns name	Used to assign a "name" to a condition thread. Source is a condition node, target can be any service.
2	Leaf	PREV	has previous instance	A relationship in which the target act is a predecessor instance to the source act. Generally each of these instances is similar, but no identical. In healthcare coverage it is used to link a claim item to a previous claim item that might have claimed for the same set of services.
2	Leaf	REFR	refers to	A relationship in which the target act is referred to by the source act. This permits a simple reference relationship that distinguishes between the referent and the referee.
2	Leaf	REFV	has reference values	Reference ranges are essentially descriptors of a class of result values assumed to be "normal", "abnormal", or "critical." Those can vary by sex, age, or any other criterion. Source and target are observations, the target is in criterion mood. This link type can act as a trigger in case of alarms being triggered by critical results.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	SUBJ	has subject	Relates an Act to its subject Act that the first Act is primarily concerned with. Examples 1. The first Act may be a ControlAct manipulating the subject Act 2. The first act is a region of interest (ROI) that defines a region within the subject Act. 3. The first act is a reporting or notification Act, that echos the subject Act for a specific new purpose. Constraints An Act may have multiple subject acts. Rationale The ActRelationshipType "has subject" is similar to the ParticipationType "subject", Acts that primarily operate on physical subjects use the Participation, those Acts that primarily operate on other Acts (other information) use the ActRelationship.
2	Leaf	SUMM	summarized by	An act that contains summary values for a list or set of subordinate acts. For example, a summary of transactions for a particular accounting period.
1	**S: ActRelationshipSequel**	SEQL	is sequel	An act relationship indicating that the source act follows the target act. The source act should in principle represent the same kind of act as the target. Source and target need not have the same mood code (mood will often differ). The target of a sequel is called antecedent. Examples for sequel relationships are: revision, transformation, derivation from a prototype (as a specialization is a derivation of a generalization), followup, realization, instantiation.
2	**S: ActRelationshipExcerpt**	XCRPT	Excerpts	The source is an excerpt from the target.
3	Leaf	VRXCRPT	Excerpt verbatim	The source is a direct quote from the target.
2	**S: ActRelationshipFulfills**	FLFS	fulfills	The source act fulfills (in whole or in part) the target act. Source act must be in a mood equal or more actual than the target act.
3	Leaf	OCCR	occurrence	The source act is a single occurrence of a repeatable target act. The source and target act can be in any mood on the "completion track" but the source act must be as far as or further along the track than the target act (i.e., the occurrence of an intent can be an event but not vice versa)..
3	Leaf	OREF	references order	Relates either an appointment request or an appointment to the order for the service being scheduled.
3	Leaf	SCH	schedules request	Associates a specific time (and associated resources) with a scheduling request or other intent.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	**S: ActRelationshipReplacement**	RPLC	replaces	A replacement source act replaces an existing target act. The state of the target act being replaced becomes obselete, but the act is typically still retained in the system for historical reference. The source and target must be of the same type.
3	Leaf	SUCC	succeeds	A new order that adds to, but does not completely replace its predecessor.
2	Leaf	APND	is appendage	An addendum (source) to an existing service object (target), containing supplemental information. The addendum is itself an original service object linked to the supplemented service object. The supplemented service object remains in place and its content and status are unaltered.
2	Leaf	DOC	documents	The source act documents the target act.
2	Leaf	ELNK	episodeLink	Expresses an association that links two instances of the same act over time, indicating that the instance are part of the same episode, e.g. linking two condition nodes for episode of illness; linking two encounters for episode of encounter.
2	Leaf	GEN	has generalization	The generalization relationship can be used to express categorical knowledge about services (e.g., amilorid, triamterene, and spironolactone have the common generalization potassium sparing diuretic).
2	Leaf	GEVL	evaluates (goal)	A goal-evaluation links an observation (intent or actual) to a goal to indicate that the observation evaluates the goal. Given the goal and the observation, a "goal distance" (e.g., goal to observation) can be "calculated" and need not be sent explicitly.
2	Leaf	INST	instantiates (master)	Used to capture the link between a potential service ("master" or plan) and an actual service, where the actual service instantiates the potential service. The instantiation may override the master's defaults.
2	Leaf	MTCH	matches (trigger)	A trigger-match links an actual service (e.g., an observation or procedure that took place) with a service in criterion mood. For example if the trigger is "observation of pain" and pain is actually observed, and if that pain-observation caused the trigger to fire, that pain-observation can be linked with the trigger.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
2	Leaf	OPTN	has option	A relationship between a source Act that provides more detailed properties to the target Act. The source act thus is a specialization of the target act, but instead of mentioning all the inherited properties it only mentions new property bindings or refinements. The typical use case is to specify certain alternative variants of one kind of Act. The priorityNumber attribute is used to weigh refinements as preferred over other alternative refinements. Example: several routing options for a drug are specified as one SubstanceAdministration for the general treatment with attached refinements for the various routing options.
2	Leaf	REV	reverses	A relationship between a source Act that seeks to reverse or undo the action of the prior target Act. Example: A posted financial transaction (e.g., a debit transaction) was applied in error and must be reversed (e.g., by a credit transaction) the credit transaction is identified as an undo (or reversal) of the prior target transaction. Constraints: the "completion track" mood of the target Act must be equally or more "actual" than the source act. I.e., when the target act is EVN the source act can be EVN, or any INT. If the target act is INT, the source act can be INT.
2	Leaf	UPDT	updates (condition)	A condition thread relationship specifically links condition nodes together to form a condition thread. The source is the new condition node and the target links to the most recent node of the existing condition thread.
2	Leaf	XFRM	transformation	Used when the target Act is a transformation of the source Act. (For instance, used to show that a CDA document is a transformation of a DICOM SR document).

ActStatus (used in RIM attribute: Act.statusCode)

This table controls values for structural elements of the HL7 Reference Information Model. Therefore, it is part of the Normative Ballot for the RIM.

Lv	Type, Domain name and/or code	Mnemonic	Print Name	Definition/Description
1	**S: ActStatusNormal**	normal	normal	Encompasses the expected states of an Act, but excludes "nullified" and "obsolete" which represent unusual terminal states for the life-cycle.
2	Leaf	aborted	aborted	The Act has been terminated prior to the originally indended completion
2	Leaf	active	active	The Act can be performed or is being performed
2	Leaf	cancelled	cancelled	The Act has been abandoned before activation.
2	Leaf	completed	completed	An Act that has terminated normally after all of its constituents have been performed.

