1. edition

November 2006

Comparing GEPJ and HL7 V3

DANSK STANDARD
Danish Standards

Kollegievej 6
DK-2920 Charlottenlund
Tel: +45 39 96 61 01
Fax: +45 39 96 61 02
dansk.standard@ds.dk
www.ds.dk

Table of Contents

1 Executive summary

The objective of this paper is to answer the following three questions:

1. Are GEPJ and HL7 comparable regarding objectives and fundamental characteristics?

2. Can GEPJ be mapped into HL7 without loss of clinical information?

3. Can HL7 be mapped into GEPJ without loss of clinical information?

This is not an in-depth comparison of GEPJ to HL7, but meant as an initial exploration of the compatibility of the two models.

The comparison performed in this project has been a technical comparison only. Technical conclusions are drawn from the comparison and no political or strategic conclusions are intended.

The objectives of GEPJ and HL7 differ. GEPJ is used to define the clinical information in an EHR system while HL7 has a broader objective. HL7 is used to define the interfaces between different types of healthcare information systems. This difference aside, what is compared are the HL7 exchange of complete records from EHR systems and the GEPJ model.

Generally the basic concepts match well. Some clinically relevant concepts in GEPJ are not present in HL7, it is envisaged that these concepts over time could be incorporated in HL7 for a more comprehensive model. The opposite is also the case which suggests a basis for convergence from both standards.

History is handled differently in the two standards. GEPJ requires a complete history, as current data is linked to older data. HL7 does not have the same requirement, as it handles snapshots of information intended for exchange at a given point in time. The implications of the difference in history handling have not been investigated in depth and should be looked at if concrete interoperability wishes emerge.

GEPJ does not need further refinement before use. HL7 on the other hand, is meant as an international applicable standard and therefore requires national profiling.

2 Introduction

GEPJ version 2.2 (GEPJ hereafter) is the current Danish national conceptual model for documentation of clinical information in the Electronically Health Record (EHR). Being developed as a Danish model, there is an interest in investigating the standards convergence to international standards.

HL7 version 3 (HL7 hereafter) is an international standard for interoperability between healthcare information systems. As HL7 is considered a fairly mature standard and is an internationally recognized American standard, it has been selected for comparison with GEPJ in order to explore the GEPJ model's ability to co-exist with international standards.

On the initiative of the standardization committee (SUSI) a taskforce was established and this project has been completed with participants from Danish Standards (DS), the National Board of Health (SST) and four suppliers of EHR systems: CSC, Systematic, IBM/Acure and WM-data. In this document the results of the work of this project is presented.

The intent of the comparison has been to answer the questions:

1. Are GEPJ and HL7 comparable regarding objective(s) and fundamental characteristics?

2. Can GEPJ be mapped into HL7 without loss of clinical information?

3. Can HL7 be mapped into GEPJ without loss of clinical information?

The questions have been answered by exploring the extent of the intersection of the GEPJ concepts and the HL7 concepts:

- By comparing fundamental characteristics in HL7 and GEPJ.

  Objectives behind the models and other fundamental characteristics have been compared.

- By investigating to what extent it is possible to express the intent of the concepts in GEPJ in terms of HL7.

  The investigation has been performed by mapping the GEPJ concepts into HL7 D-MIMs and R-MIMs.

- By investigating to what extent it is possible to fit the information in HL7 into the concepts of the GEPJ model.

  This has been done by mapping two examples of R-MIMs into the GEPJ model.

This is not an in-depth comparison of GEPJ to HL7, but meant as an initial exploration of the compatibility of the two models.

The comparison performed in this project has been a technical comparison only. Technical conclusions are drawn from the comparison and no political or strategic conclusions are intended.

This document is intended for persons who are interested in the relationship between GEPJ and HL7.

Executive summary, the introduction and parts of the fundamental characteristics section can be read without technical knowledge about HL7 and GEPJ. The technical parts are summarized for non-technical decision makers in the executive summary. The introduction gives the background for the project.

In the rest of this report the reader is assumed to have basic knowledge of GEPJ and HL7. These sections provide the reader with a more technical understanding of the mapping between the HL7 and the GEPJ concepts.

3 Fundamental Characteristics

In the following, the fundamental characteristics of GEPJ and HL7 are compared.

3.1 GEPJ Objective

The objective of the GEPJ model is defined as follows:

> GEPJ is an acronym for Basic structure for Electronic Health Records (in Danish). GEPJ specifies requirements for clinical documentation in an electronic health record, EHR. The starting point of the specification is the individual steps of the clinical work. GEPJ specifies the structure, relationships, and formalized data that are necessary for providing cohesive documentation system. GEPJ is multi disciplinary oriented around the patient's problem and is capable of handling health care threads across sector boundaries. GEPJ is not a specification for the construction of a record, but a specification of requirements for clinical documentation in an EHR.

Source: http://www.sst.dk/gepj

3.2 HL7 Objective

The objective of the HL7 standard is defined as follows:

> HL7's mission is to provide standards for the exchange, management and integration of data that supports clinical patient care and the management, delivery and evaluation of healthcare services. Specifically, to create flexible, cost effective approaches, standards, guidelines, methodologies, and related services for interoperability between healthcare information systems.

Source: HL7 Version 3 Standard, Introduction, Package Note to Readers, HL7 Organizational Overview, Mission

3.3 Comparing the Model Characteristics

3.3.1 Comparison of the Objectives

Clearly, the objectives of GEPJ and HL7 differ. GEPJ focuses on capturing the clinical information related to the patient (the state of the patient) and only to a limited extent documents the processes that generate the information. The aim of GEPJ is to define the clinical information that an EHR system must be able to capture about a patient with regard to clinical decision making.
HL7 has a broader objective and deals with both the healthcare processes and the many types of data exchanged between healthcare information systems. HL7 is used to define the interfaces that different types of healthcare information systems (actors in the healthcare domain) must be able to handle, e.g. the interfaces of a pharmacy information system.
Hence, making a one-to-one mapping between the two models is like comparing apples and oranges. What can be compared are the HL7 exchange of complete records from EHR systems and the GEPJ model. Because GEPJ mainly deals with clinical data and little with the surrounding processes, exchange of non-clinical data cannot be mapped into GEPJ and data related to the clinical process as such can only be mapped into GEPJ to a limited extent.

3.3.2 Comparing the National Applicability

GEPJ is a national model based on the Danish healthcare professional's modus operandi. It does not need further refinement before use. HL7 on the other hand, is meant as an internationally applicable standard and therefore requires national profiling. If HL7 is going to be used for messaging a national adaptation has to be done. This would be similar to the work done by the Danish organization Medcom regarding the international EDIFACT standards.

3.3.3 Comparing the Object Models

The two object models differ. This means that the GEPJ model translated to HL7 has a different number of objects and different relations between these objects. If such a model was to be translated back into GEPJ, this would require a very clear semantic description.

3.3.4 Comparing the Handling of Object History

There is a difference in the way the models handle object history. By nature, HL7 handles exchange of information – snapshots of the information at a given point in time. HL7 can describe a model with explicit history (as shown in Condition Tracking), but it seems like an 'add-on' and does not feel like a natural description. GEPJ requires a complete history, as current data is linked to older data in GEPJ.
The differences may have implications if GEPJ and HL7 systems are integrated. Even small use cases involving a few objects will end up with a very different collection of objects. This will make it difficult for GEPJ systems and HL7 systems to make references to each other.
The implications of the difference in history handling have not been investigated further, but should be looked at if concrete interoperability wishes emerge.

In the following, the details about how the models handle object history are described. This is only aimed at technically interested readers.

GEPJ Object History

In GEPJ, all concepts are represented as two classes; one class that describes the static (un-modifiable) attributes and one class that describes the attributes that may change over time. GEPJ has no term for the combination of the two.

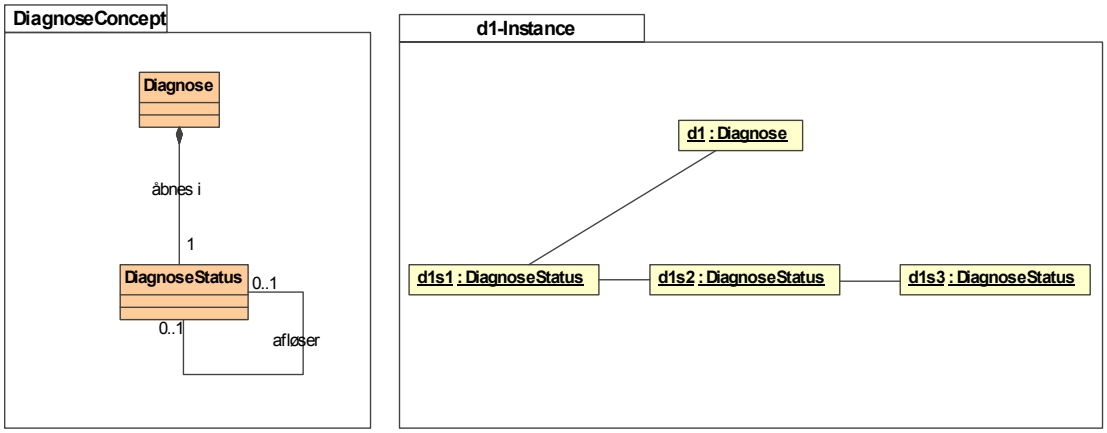


Figure 1 Left: A simplified model of a Diagnosis. Right: The instances created after a few updates. The last status instance is the actual value. The previous state instances are the history.

Objects in GEPJ are immutable. Changes are made by creating new instances of status objects and relate them to the previous ones. At any given point in time, the value is the combination of the main object and one single state object.

The most important associations in the GEPJ model are expressed by association classes. Here they are called GEPJ relations (because they correspond well to HL7 ActRelations). GEPJ relations are first-class concepts and can be created without modifying the endpoints. Instances of relations have state objects as well and can be updated by creating new relation status instances.

Relations can go to/from main objects and/or status objects.

This design ensures an **explicit history** of all information. This means that a system based on GEPJ can implement a complete history by simply storing all objects. It also means that a system needs the complete history to have a consistent model.

All main objects and all status objects have individual object identifiers. This means that a diagnosis has many identifiers; one for the main object and one for every update.

The business lifecycle of an object (including relations) is described as a state diagram. Changing the state of an object is a modification and consequently gives a new status instance.

Objects can not be deleted. If false information (for any reason) enters the system it can be replaced by corrected information. The false information remains in the model and is marked as “Rettet” with a relation to the correct data or to itself if no replacement exists.

Obsolete information is indicated by the state value.

HL7 Object History

Concepts in HL7 are described as subclasses (restrictions) of a few classes in the reference model RIM. Most important is the class Act. Acts are documentation of actions performed or planed.

In general objects are mutable. It is defined individually for each attribute weather it can be modified or not.

In general, objects can be updated and they do not contain version information. It is up to the individual systems to decide if historical information is needed and how it should be implemented. The **history is implicit** and must be deduced from timestamps or other means. A system can store as much historical information as needed and as complete as needed.

The business lifecycle of a concept in HL7 is described by a combination of moods and states. Moods define major steps, i.e. going from planning to execution. State describes the progress within a given mood, i.e. going from ‘in progress’ to ‘done’. State is changed by updating an existing instance. Mood is changed by creating a new instance since the attributes, associations and business rules are defined by mood.

Objects can effectively be deleted by changing the state to ‘nullified’.

State change can optionally be described by ControlActs. ControlActs are wrappers to Acts that describe a change in state.

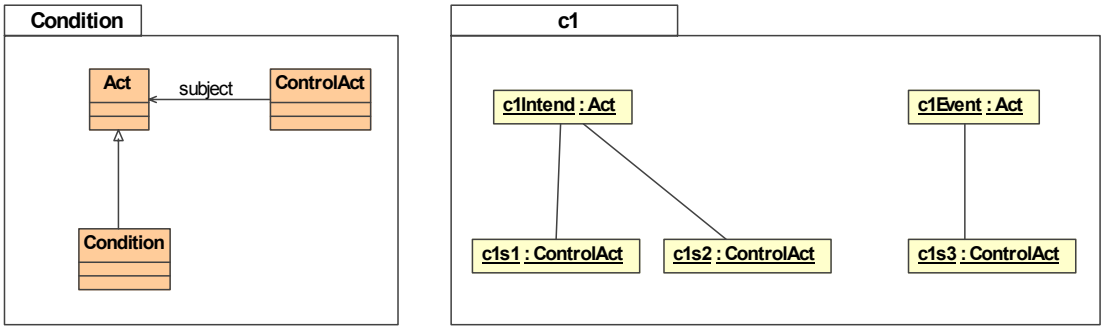


Figure 2 Left: Simplified model equivalent to the one in figure1. HL7 restrictions are special kinds of specializations. Right: The instances created after changing mood and state.

By using Control Acts it is possible to make explicit history in the model. This approach is taken in the Condition Tracking domain [3]. Control Acts are acts that have an identifier. Since the Control Act is related to a specific state transition, the combination of act identifier and control identifier can be used as the identifier of an act at a specific point in time.

Relations in HL7 are made by instances of ActRelationship sub-classes. Relations are not first-class objects. They are “attributes” of the source act, i.e. creating a relation is an update of the source act. This indicates that a HL7 system will have to handle more versions than a GEPJ system if it needs full history.

Clinical relations are between real acts, not Control Acts. (In contrast to GEPJ where you can relate to a state object).

Since the general state diagram for Act allows a “Revise” transaction from almost any state to itself, Control Acts can be used to document the history of almost any modification.

3.4 Conclusion

The objectives of GEPJ and HL7 differ. GEPJ is used to define the clinical information in an EHR system while HL7 has a broader objective. HL7 is used to define the interfaces between different types of healthcare information systems.
This difference aside, what is compared are the HL7 exchange of complete records from EHR systems and the GEPJ model.

GEPJ does not need further refinement before use. HL7 on the other hand, is meant as an international applicable standard and therefore requires national profiling

History is handled differently in the two standards. GEPJ requires a complete history, as current data is linked to older data. HL7 does not have the same requirement, as it handles snapshots of information intended for exchange at a given point of time.
This difference will make it difficult for GEPJ systems and HL7 systems to make references to each other.

4 Basic GEPJ Concepts

4.1 Introduction

This chapter addresses the possibility of expressing the basic clinical concepts from GEPJ in HL7. That is, the classes Forloeb (Health Care Thread), Diagnose (Diagnosis), Intervention, Interventionsresultat (Result), OperationeltMaal (Operational Goal), EvalueringsResultat (Evaluation Result) and supporting classes and relations.

Each of the classes is analysed individually. The examination focuses on the Care Provision D-MIM from the May 2006 ballot.

4.2 Diagnosis

From the HL7 definition of Observation:

> 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.
>
> 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.

The HL7 definition of Condition:

> An observable finding or state that persists over time and tends to require intervention or management and is, therefore, distinguished from an observation made at a point in time.

A Condition, as a subclass of Observation, plays the role of diagnosis receptacle in the Care Provision D-MIM. This is the place to register the diagnosis of an on-going condition that is the focus of the care of the patient. The actual diagnosis is to be found in the value field of the Observation.

In contrast, GEPJ treats the diagnosis as a first-class object in itself, but does not register the Act of establishing or tracking the diagnosis (which means, for instance, that you cannot register the intent of establishing a diagnosis in GEPJ). The way for us to map a Diagnose object to HL7 is to create a Condition in the ‘event’ mood with the proper observation code. The actual diagnosis from the ‘art’ attribute is put into the value field of the observation as a coded value.
In addition to the ‘art’ attribute, the ‘Diagnosestatus’ object holds a number of attributes that are relevant to the diagnosis as well. These attributes are:

Attribute	Description	HL7 mapping
besluttet/besluttetAf	The time of and person responsible for the existence of the information. This	This is mapped to the performer role in HL7.

	is typically the doctor that determined the diagnosis.	
dokumenteret/dokumenteretAf	The time of and person responsible for the registration of data in the system.	This is mapped to the dataEnterer role in HL7.
diagnostiknotat	A note by the person responsible for the existence of the information. (Technically, these are separate objects associated to the 'diagnosestatus'.)	In HL7, this is mapped to an Annotation object associated with the condition.

In GEPJ, the diagnoses are arranged in a diagnosis network through a number of relations on the diagnoses. These relations are of course ActRelationships in HL7. The GEPJ relations are as follows.

GEPJ diagnosis relation	**Description**	**HL7 mapping**
DAarsag	This relation represents a cause-and-effect relationship between two diagnoses. For instance, diabetic ulcer as a cause of diabetes.	This relation is mapped to a HL7 component (COMP) relation between the two conditions that correspond to the diagnoses.
DDiff	This relation represents a diagnosis that is suggested as an alternative ("second opinion") to another diagnosis.	This relation cannot be represented in HL7 directly.
DLoesKobling	This relation indicates that two diagnoses are "loosely coupled" – that is, it has been decided that there is an unspecified relationship between the diagnoses.	This relation cannot be represented in HL7 directly.
DKvalificering	This relation represents that a diagnosis replaces another because the	This corresponds directly to a HL7 replace (RPLC) relation.

	new diagnosis is a better representation of the current opinion on the patient's state.	
DKomplicering	This relation indicates that a diagnosis is a complication of a disease.	HL7 does not distinguish between complications and other cause-and-effect relationships, so this relationship is mapped to a component (COMP) as well.

In GEPJ, a number of reasons can be given for the conclusion that a condition exists (indeed, at least one reason *must* be given). These so-called 'fokuserede oplysninger' are modelled as references to any piece of information from the EHR. The corresponding Care Provision relation is the Support ActRelationship between ConditionNode and CareStatement.

4.3 Intervention

The GEPJ class Intervention is a versatile class. The scope is broader than the name suggests and in recent versions of GEPJ the class has sometimes been referred to simply as "Health activity." This name reveals the true breadth of the scope of the intervention class. It is defined as any activity performed by health care individuals, directly or indirectly directed towards one single patient, with a health care related goal in mind.

An intervention can be any of the following:

- An operation.
- A measurement.
- Obtaining a specimen for further analysis.
- Analyzing a specimen.
- Any treatment of a patient's ailments (i.e. physiotherapy).
- A higher-level activity consisting of several basic activities (like a surgical procedure consisting of activities like anaesthesia, incision, suture, and others).

In addition the model allows for the following kinds of interventions, which are modelled using subclasses of Intervention and will not be treated here:

- PatientEncounter (PatientKontakt)
- MedicineOrdination (Medicinering)
- GivingMedicine (MedicinUdlevering).

Seeing the list of different roles that an Intervention can play in the GEPJ model, we can observe that it corresponds closely to the HL7 concept of a CareEntry. As expected, a CareEntry in HL7 corresponds to a wide range of concrete classes. This means that we must map an Intervention to different classes depending on its context.

In the typical case, an Intervention maps to an Act, as this is the only HL7 class broad enough to capture the different roles that an Intervention can play in GEPJ. The only exception is for Interventions that acts as a container within the hierarchy, since a

constraint on Act forbids that the Act takes the role of a container. Instead, higher-level activities will be mapped to an Organizer.

In GEPJ, all interventions are given a reason (Indikation) in the form of a diagnosis. This is the GEPJ way of making a problem-oriented medical record (POMR). The Reason relation in HL7 corresponds to the GEPJ concept of an Indikation. The Reason relation traces through a CareProvision, which states that a set of CareEntries are planned for the same reason. We generate such a CareProvision in our mapping, since this information is not expressed explicitly in GEPJ.

4.4 Interventionsstatus

The Interventionsstatus class from GEPJ deserves special mentioning because of its complex relationship with HL7. GEPJ has a single concept defining both the state and mood of an Act. This means that sometimes a simple state change in GEPJ will result in changing the mood of the corresponding HL7 act, and hence the specification of HL7 will result in a new object. In this way an Intervention with several changes may result in a number of different HL7 Acts each with a different mood and each possibly going through different state changes themselves.

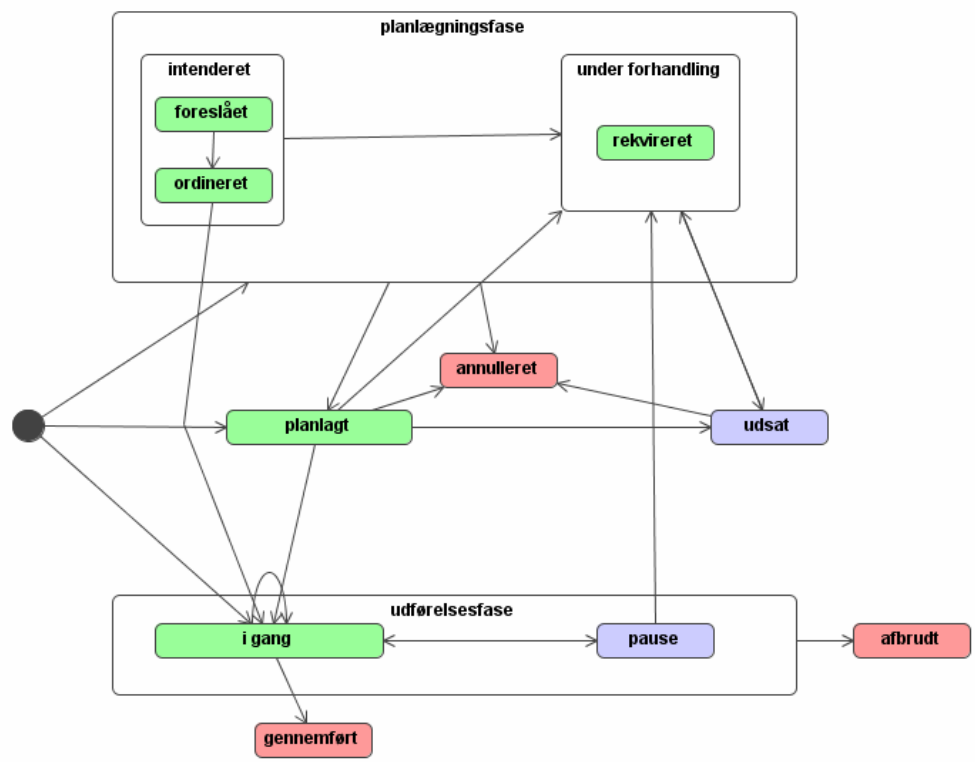


Figure 3 The state diagram for an Intervention in GEPJ

The state diagram in Figure 3 above represents the possible state transitions for an Intervention in GEPJ. Below is a table showing the mapping corresponding to each of the GEPJ states.

GEPJ state	Mapping
'Foreslået' (suggested)	An intervention in the state 'Foreslået' corresponds to an Act in the mood 'recommendation'. This state may change into 'annulleret' which corresponds to changing the state of the

	Act to ‘cancelled’, or it may change to ‘ordineret’, ‘rekvireret’, ‘planlagt’, or ‘i gang’, each of which require a new Act in a different mood.
‘Ordineret’ (prescribed)	An Intervention in the state ‘Ordineret’ corresponds to an Act in the mood ‘promise’. This state may change into ‘annulleret’ which corresponds to changing the state of the Act to ‘cancelled’, or it may change to ‘rekvireret’, ‘planlagt’, or ‘i gang’, each of which require a new Act in a different mood.
‘Rekvireret’ (requested)	An Intervention in the state ‘Rekvireret’ corresponds to an Act in the mood ‘request’. This state may change into ‘annulleret’ or ‘udsat’ which correspond to changing the state of the Act to ‘cancelled’ and ‘held’, respectively, or it may change to ‘planlagt’, or ‘i gang’, each of which require a new Act in a different mood.
‘Planlagt’ (planned)	An Intervention in the state ‘Planlagt’ corresponds to an Act in the mood ‘appointment’. This state may change into ‘annulleret’ or ‘udsat’ which correspond to changing the state of the Act to ‘cancelled’ and ‘held’, respectively, or it may change to ‘rekvireret’, or ‘i gang’, each of which require a new Act in a different mood.
‘I gang’ (active)	An Intervention in the state ‘I gang’ corresponds to an Act in the mood ‘event’. This state may change into ‘pause’, ‘afbrudt’ or ‘gennemført’ which correspond to state changes to ‘suspended’, ‘aborted’, and ‘completed’, respectively.
‘Udsat’ (postponed)	An Intervention in the state ‘Udsat’ corresponds to any Act in the state ‘held’. The mood of the Act is determined by the former state of the Intervention.
‘Annulleret’ (cancelled)	An Intervention in the state ‘Annulleret’ corresponds to any Act in the state ‘cancelled’. The mood of the Act is determined by the former state of the Intervention.
‘Pause’ (paused)	An Intervention in the state ‘Pause’ corresponds to any Act in the state ‘suspended’. The mood of the Act is determined by the former state of the Intervention.
‘Afbrudt’ (aborted)	An Intervention in the state ‘Afbrudt’ corresponds to any Act in the state ‘aborted’. The mood of the Act is determined by the former state of the Intervention.
‘Gennemført’ (completed)	An Intervention in the state ‘Gennemført’ corresponds to any Act in the state ‘completed’. The mood of the Act is determined by the former state of the Intervention.

The above table covers the issue of which state to give the GEPJ states corresponding to moods rather than to states. This matter is determined by the legal state transitions for an Act depicted in Figure 4 below. Since the GEPJ states ‘Foreslået’, ‘Ordineret’, ‘Rekvireret’, and ‘Planlagt’ need to be able to legally change state to ‘cancelled’ and ‘held’ they need to be in the state ‘new’. The GEPJ state ‘I

gang' needs to be able to change to 'aborted', 'suspended', and 'completed', and hence must be in the state 'active'.

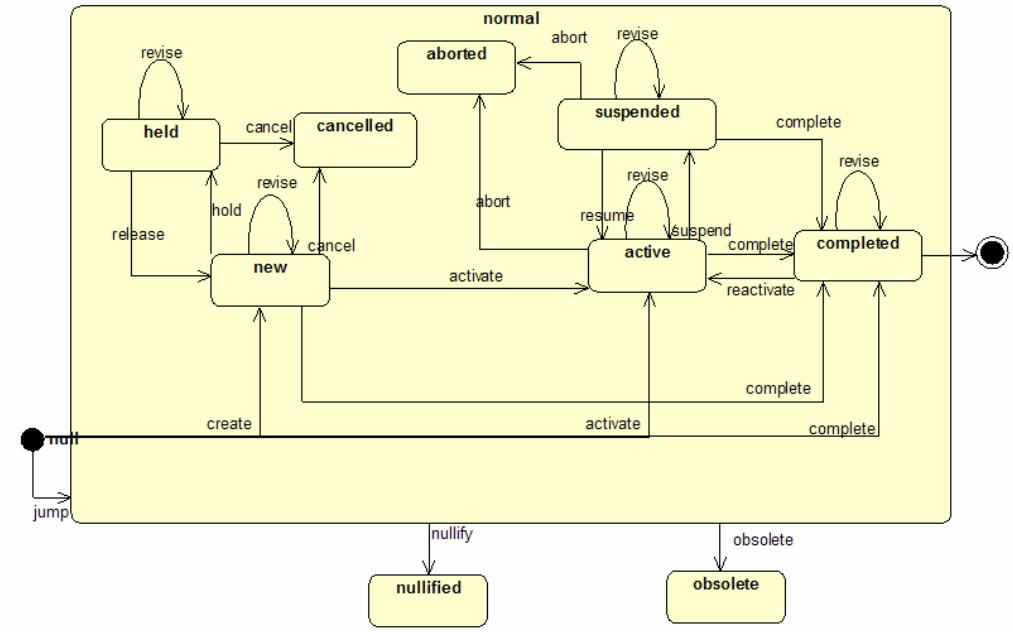


Figure 4 the state diagram for an Act in HL7

4.5 Interventionsresultat (Result)

Like for Diagnose, an Interventionsresultat is not treated like a first-class object in HL7, but rather as a value field in a corresponding Observation. Through the results of the previous sections on Interventions we should expect the existence of an Act in the mood 'event' when a Result is present (since a corresponding Intervention must have been completed). The value of the Result can be put into the value field of this Act.

Now, the value of the Interventionsresultat (the ResultInfo class) is complex in GEPJ, as it allows building a hierarchical value for the result of a piece of information. We map this to HL7 by building a hierarchy of Organizers and Observations using the Component relation. Thus, we map a RInfoContainer to an Organizer and a RInfoElement to an Observation.

4.6 OperationeltMaal (Operational Goal)

The HL7 definition of the mood 'goal':

Expectation to make a specific observation with a desired value at a predefined future time

This is exactly what the GEPJ class OperationeltMaal is designed to solve. So an operational goal is an observation in 'goal' mood.

In GEPJ the operational goal contains an interval for the expected outcome of a measurement. This is modelled in HL7 using an ObservationRange.

4.7 EvalueringsResultat

HL7 does not seem to have a concept corresponding directly to the evaluation of whether an observation meets its goal, but it is clearly an Observation in the ‘event’ mood.

4.8 Forloeb (Health Care Thread)

In GEPJ the class ‘Forloeb’ comprises the top level of the hierarchies. In particular, it pinpoints a diagnosis in the diagnosis hierarchy as the ‘forloebsdiagnose’ which is the primary diagnosis in that particular health care thread. This corresponds conceptually to the HL7 concept of an ‘Episode of Care’ which is tracked through a number of condition nodes corresponding to changing conditions throughout the episode. This means that we map a Forloeb with changing diagnosis related by DKvalificering to a series of Condition Nodes related with an Elink (ELNK) relationship.

4.9 Conclusion

Mapping the basic GEPJ concepts to HL7 is complex but essentially possible for the most part. The only problem that gives rise to difficulties is the lack of HL7 relations corresponding to DDiff and DLoesKobling.

The implication from a clinical perspective is of some significance with regard to DDiff. It is envisaged that this concept over time could be incorporated in HL7 for a more comprehensive model.

5 Medication

5.1 Introduction

This chapter addresses the possibility of expressing the medication related information item in GEPJ in HL7.

Medication in GEPJ is modelled as a specialization of the generic class Intervention. Thus this chapter will not treat the super-classes that are common for all interventions (please refer to the chapter on intervention).

5.2 Medication in GEPJ

In the UML documentation in GEPJ the medication domain is modelled in a single class diagram: “Medicinering” (Figure 5). The model contains two major classes “Medicinering” (Medication order/prescription) and “Medicinudlevering” (giving medicine). “Medicinering” is an abstract class that has three specializations: “SkemaMedicinering” (medication according to schema), “FastMedicinering” (fixed medication) and “VariabelMedicinering” (flexible medication).

Furthermore the model contains some supporting classes: “Medicinblanding” (mixed medicine) and “Ingediens” (ingredient).

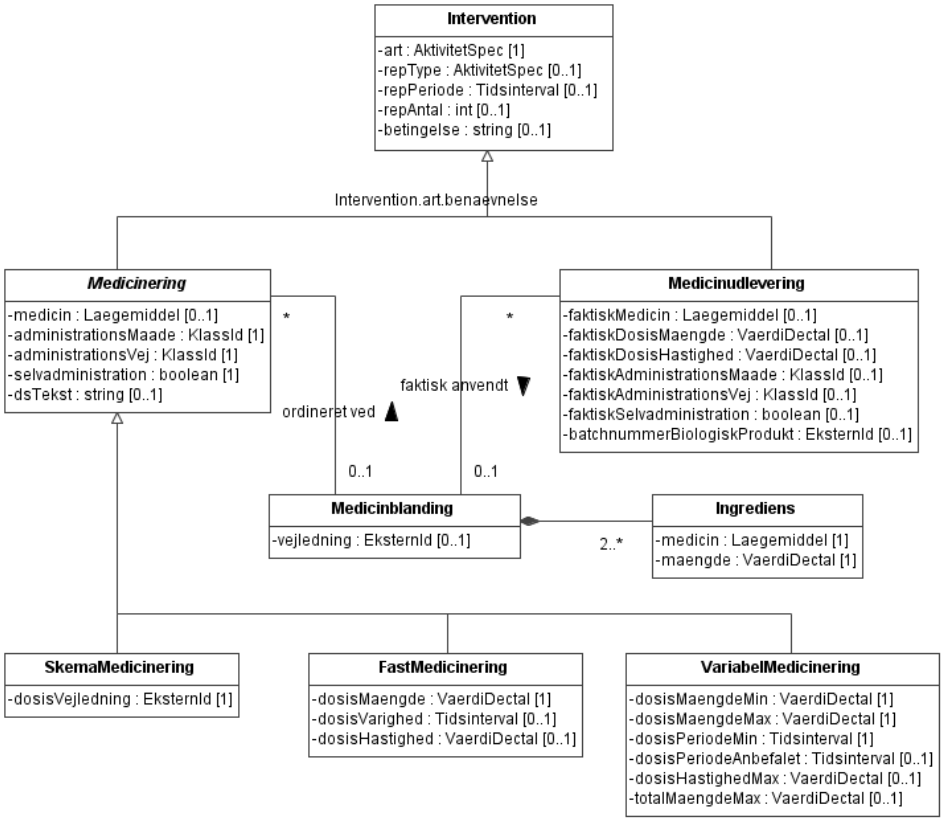


Figure 5: GEPJ class diagram "Medicinering"

5.3 Medication in HL7

In HL7 the model for clinical medication can be found in the Domains>Health and Clinical Management Domains>Pharmacy section. Information on the drugs/medicine used is in the Domains>Health and Clinical Management Domains>Medicine section.

The D-MIM Pharmacy Domain Model (PORX_DM000000) (Figure 6) shows the corresponding modelling to the GEPJ medication domain model (Figure 5).

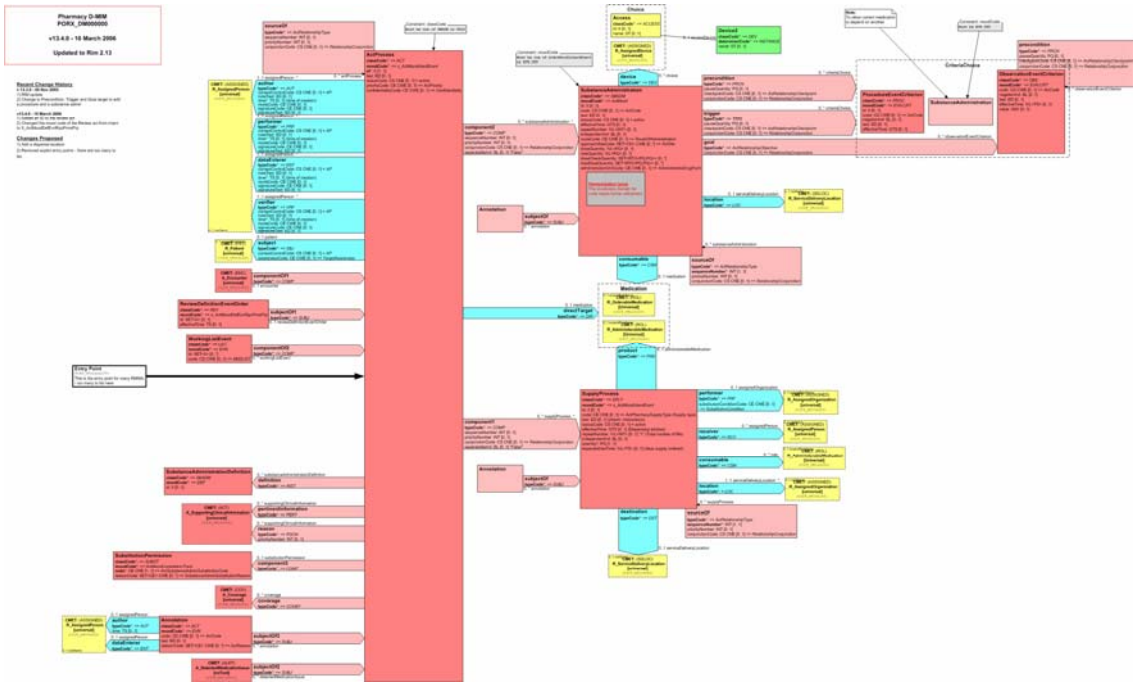


Figure 6: Pharmacy Domain Model HL7

The Pharmacy D-MIM contains the SubstanceAdministration Class and related classes:

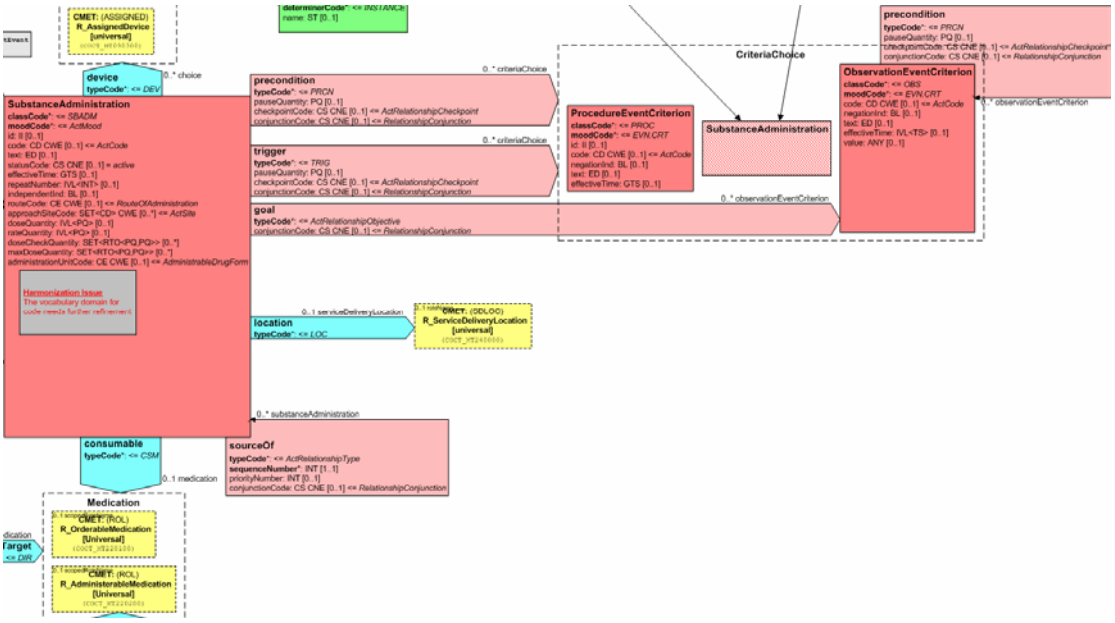


Figure 7: SubstanceAdministration Class and related classes

The SubstanceAdministration Class and its attributes can be seen in Figure 8

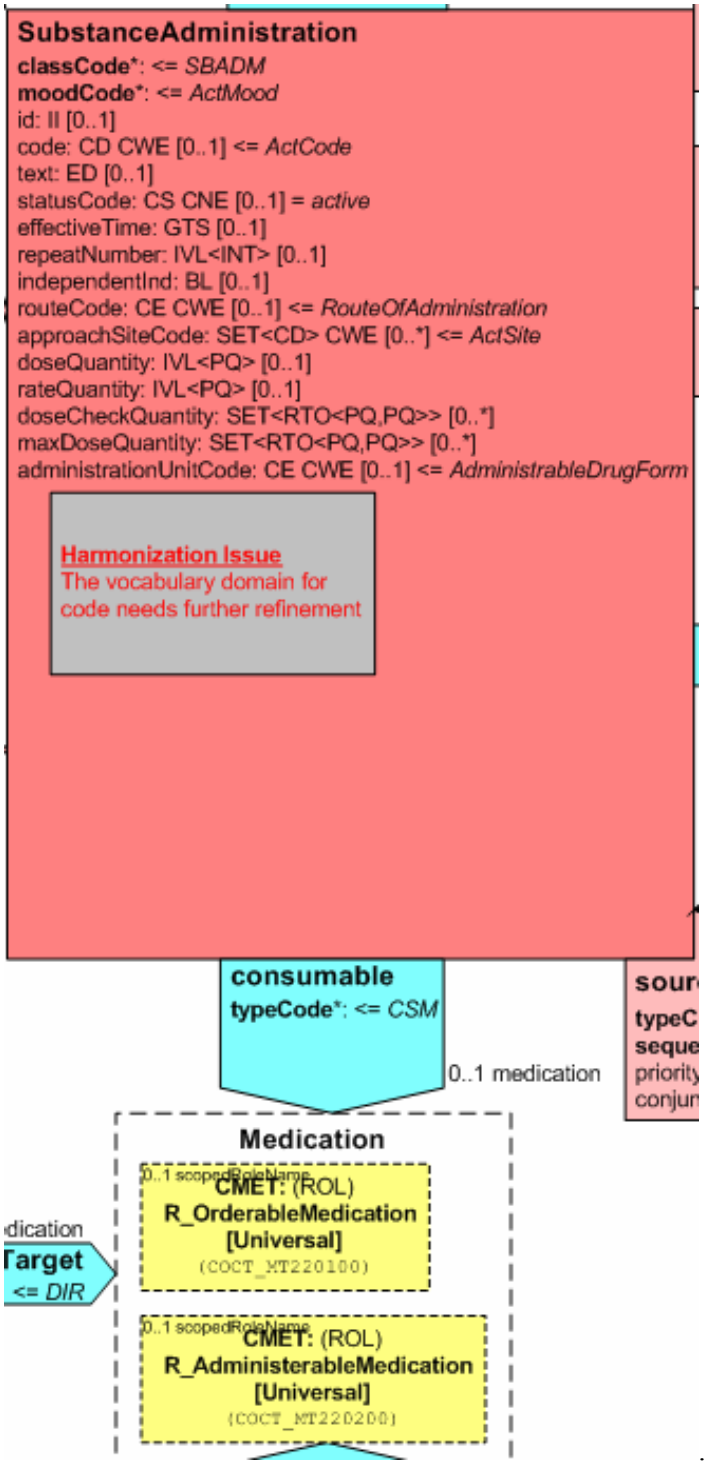


Figure 8: SubstanceAdministration Class and its attributes

5.4 Comparison of GEPJ and HL7 classes and attributes

GEPJ element **Classes**	**Documentation**	**Corresponding HL7 element**
Medicinudlevering	Specifies the medication given. The duration can be calculated from the associated instances of 'Interventionstatus'	Pharmacy D-MIM PORX_DM000000 SubstanceAdministration Mode event = event
Medicinblanding	Specifies a individual patient related mixture and medicine – e.g. typically inotropic and peadiatric infusions are mixed individually	C-MET R_AdministerableMedication (COCT_RM220200) Medicine via relational asMedication
Ingrediens	Specifies a component in a mixture of medicine	C-MET R_AdministerableMedication (COCT_RM220200) Ingredient via an ingredient Substance
VariabelMedicinering	Specifies a medication with no pre-defined pattern – as opposed to fixed medication (FastMedicinering).	Pharmacy D-MIM PORX_DM000000 SubstanceAdministration
FastMedicinering	Specifies a medication with a pre-defined pattern, as opposed to a variable medication (VariabelMedicinering)	Pharmacy D-MIm PORX_DM000000 SubstanceAdministration
SkemaMedicinering	Specifies that the medication must be given according to a instruction	Pharmacy D-MIm PORX_DM000000 SubstanceAdministration
Medicinering (abstract class)	A specification of a medication (prescription – medication order)	Pharmacy D-MIm PORX_DM000000 SubstanceAdministration Mode event = intent

GEPJ element Attributes	Documentation	Corresponding HL7 element Where nothing else stated source is Pharmacy D-MIM	Documentation
FastMedicinering. dosisHastighed	Specifies the rate of a medication – typically infusion rate	SubstanceAdministration.rateQuantity	This attribute is used for medicines administered continuously (e.g. gases or infusions), so that the rate of administration can be specified. Further detail on the use of the rateQuantity attribute is given in a Dosage Instructions analysis.
FastMedicinering. dosisMaengde	Specifies the quantity per giving	SubstanceAdministration.doseQuantity	The amount of the medicine to be administered (or that was administered in the EVN mood). Further detail on the use of the doseQuantity attribute is given in a Dosage Instructions analysis.
FastMedicinering. dosisVarighed	Specifies the duration of a medicine giving.	SubstanceAdministration.effectiveTime	The usage of this attribute in the substance administration act varies according to the mood - in PRMS, PRP and RQO this would hold the time that the medicine administration should take place (i.e. the timing part of the dosage instructions), in EVN, it is the time that the administration did take place. Please refer to the discussion of timing in pharmacy messaging in the Introduction and to a separate analysis and discussion of Dosage Instructions.
Ingrediens. medicin	Specifies the ingredient used	C-MET R_AdministerableMedicationSubstance.code	No documentation
Ingrediens. maengde	Specifies the quantity of an ingredient used	C-MET R_AdministerableMedicationIngidient.quantity	No documentation
Medicinblanding. Vejledning	Reference to specification regarding mixture and/or method	Medicine.description	R_OrderableMedication (COCT_RM220100)

GEPJ element Attributes	Documentation	Corresponding HL7 element Where nothing else stated source is Pharmacy D-MIM	Documentation
Medicinering.medicin	Specifies the medicine, e.g. Normal saline 1000 ml	R_OrderableMedication (COCT_RM220100) Medication.code	For entityCode: A: x_Medicine A19668: A type of Administered Substance that is manufactured from raw organic or inorganic ingredients and used in the course of a patient's therapy.
Medicinering.administrationsMaade	Specifies the way (technique) the medicine is administrated e.g. injection, infusion	SubstanceAdministration.routeCode	This attribute uses coded data for the Route of Administration information for the administration of the medicine.
Medicinering.selvadministration	Specifies if the patient administrates the medicine himself	No corresponding element in HL7 on D-MIM level	
Medicinering.dsTekst	Specifies textual information regarding the medication.	SubstanceAdministration.text	This is the full "human readable" text of the substance administration act.
Medicinering.administrationsVej	Specifies the route for the medication, e.g intravenously, oral.	SubstanceAdministration.routeCode	This attribute uses coded data for the Route of Administration information for the administration of the medicine.
Medicinudlevering.batchnummerBiologiskProdukt	Specifies a batch number for a biological product if required.	C-MET R_AdministerableMedication lotNumberText [0..1]	No documentation
Medicinudlevering. faktiskMedicin	Specifies the medicine given.	C-MET R_AdministerableMedication Medication.code	For entityCode: A: x_Medicine A19668: A type of Administered Substance that is manufactured from raw organic or inorganic ingredients and used in the course of a patient's therapy.

GEPJ element Attributes	Documentation	Corresponding HL7 element Where nothing else stated source is Pharmacy D-MIM	Documentation
Medicinudlevering. faktiskAdministrationsVej	Specifies the route for the medicine given.	SubstanceAdministration.routeCode	This attribute uses coded data for the Route of Administration information for the administration of the medicine.
Medicinudlevering. faktiskDosisHastighed	Specifies the rate by which the medicine is given.	SubstanceAdministration.rateQuantity	This attribute is used for medicines administered continuously (e.g. gases or infusions), so that the rate of administration can be specified. Further detail on the use of the rateQuantity attribute is given in a Dosage Instructions analysis.
Medicinudlevering. faktiskAdministrationsMaade	Specifies the way (technique) used when the medicine is given.	SubstanceAdministration.routeCode	This attribute uses coded data for the Route of Administration information for the administration of the medicine.
Medicinudlevering. faktiskDosisMaengde	Specifies the quantity of the medicine given.	SubstanceAdministration.doseQuantity	The amount of the medicine to be administered (or that was administered in the EVN mood). Further detail on the use of the doseQuantity attribute is given in a Dosage Instructions analysis.
Medicinudlevering. faktiskSelvadministration	Specifies if the medicine was administrated by the patient himself.	No corresponding element in HL7 on D-MIM level	
SkemaMedicinering. dosisVejledning	Instruction for a medication according to a schema.	SubstanceAdministration.annotation	This structure allows communication of particular pieces of coded information that the medication act is the subject of, such as a particular "endorsement" by the prescriber of the prescription, such that it meets local business rules and can be processed. The annotation act has two participations, author and dataEnterer, linked to the Assigned Person description, so that, if the annotation is separate from the authoring of the medication act, this can be captured and communicated here.

GEPJ element Attributes	Documentation	Corresponding HL7 element Where nothing else stated source is Pharmacy D-MIM	Documentation
VariabelMedicinering. dosisPeriodeAnbefalet	Specifies the recommended time interval between givings, e.g. 4 hours. In care of infusions this attribute states the time period from an infusion is ended until a new infusion starts.	SubstanceAdministration.effectiveTime	The usage of this attribute in the substance administration act varies according to the mood - in PRMS, PRP and RQO this would hold the time that the medicine administration should take place (i.e. the timing part of the dosage instructions), in EVN, it is the time that the administration did take place. Please refer to the discussion of timing in pharmacy messaging in the Introduction and to a separate analysis and discussion of Dosage Instructions.
VariabelMedicinering. dosisHastighedMax	Specifies the maximum dose rate for each giving, e.g. 5 mg/min.	SubstanceAdministration.rateQquantity	This attribute is used for medicines administered continuously (e.g. gases or infusions), so that the rate of administration can be specified. Further detail on the use of the rateQuantity attribute is given in a Dosage Instructions analysis.
VariabelMedicinering. dosisPeriodeMin	Specifies the minimum interval between each giving, e.g. 3 hours. In case of infusion this attribute states the period from the termination of an infusion until a new infusion starts. If continuous infusion set dosisPeriodeMin= 0.	No corresponding element found in HL7	

GEPJ element Attributes	Documentation	Corresponding HL7 element Where nothing else stated source is Pharmacy D-MIM	Documentation
VariabelMedicinering.totalMaengdeMax	Specifies the maximum quantity – either absolute or per time unit, e.g. 4000 mg, 10 litres, 300 mg/24h.	SubstanceAdministration.maxDoseQuantity	This attribute may be used to state the maximum amount of medicine that can be administered over a stated period of time; this is especially useful for limiting "as required" medications. Further detail on the use of the MaxDoseQuantity attribute is given in a Dosage Instructions analysis.
VariabelMedicinering.dosisMaengdeMax	Specifies the maximum dose per giving, e.g. 10 tablets, 50 mg. This also applies to infusions, e.g. 50 mg, 3000 ml.	SubstanceAdministration.doseQuantity	The amount of the medicine to be administered (or that was administered in the EVN mood). Further detail on the use of the doseQuantity attribute is given in a Dosage Instructions analysis.
VariabelMedicinering.dosisMaengdeMin	Specifies the minimum dose per giving, e.g. 3 tablets, 15 mg. This also applies to infusions, e.g. 5 mg, 1000 ml.	SubstanceAdministration.doseQuantity	The amount of the medicine to be administered (or that was administered in the EVN mood). Further detail on the use of the doseQuantity attribute is given in a Dosage Instructions analysis.

5.5 Conclusion

Both GEPJ and HL7 have sub-modelled the medication domain. Both models consider medication as activities (GEPJ: Intervention – HL7: act.)

The concepts that have been defined in the GEPJ model as belonging to the medication domain can be found with the same distinction in the HL7 Pharmacy D-MIM with a few exceptions. The exceptions are not considered critical with regard to the clinical content in a possible exchange of information exchange between a GEPJ based and an HL7 based documentation system.

In a few cases there could be several possibilities for mapping; consequently in case of implementing a mapping between GEPJ documentation and HL7 messages an implementation guide is recommended.

Of 7 classes all 7 are mapped.

Of 25 attributes, only three attributes related to clinical information cannot be mapped. Two attributes are related to self-medication, the third relates to minimum period between doses. The attribute related to minimum period between doses is of clinical significance and it should be investigated further if this concept should be included in HL7.

In general, we found no hindrance to express the GEPJ medication concepts in HL7 components.

6 Patient Administration

6.1 Introduction

This section contains a comparison between the patient administrative part of GEPJ and the corresponding parts of HL7.

The section starts with a description of patient administration in GEPJ and HL7. After this the individual parts from GEPJ are compared with HL7. For each class, association and association class the corresponding concepts and constructions are searched within HL7.

6.2 Patient Administration in GEPJ

Patient administration in GEPJ is illustrated with the following diagram [1]:

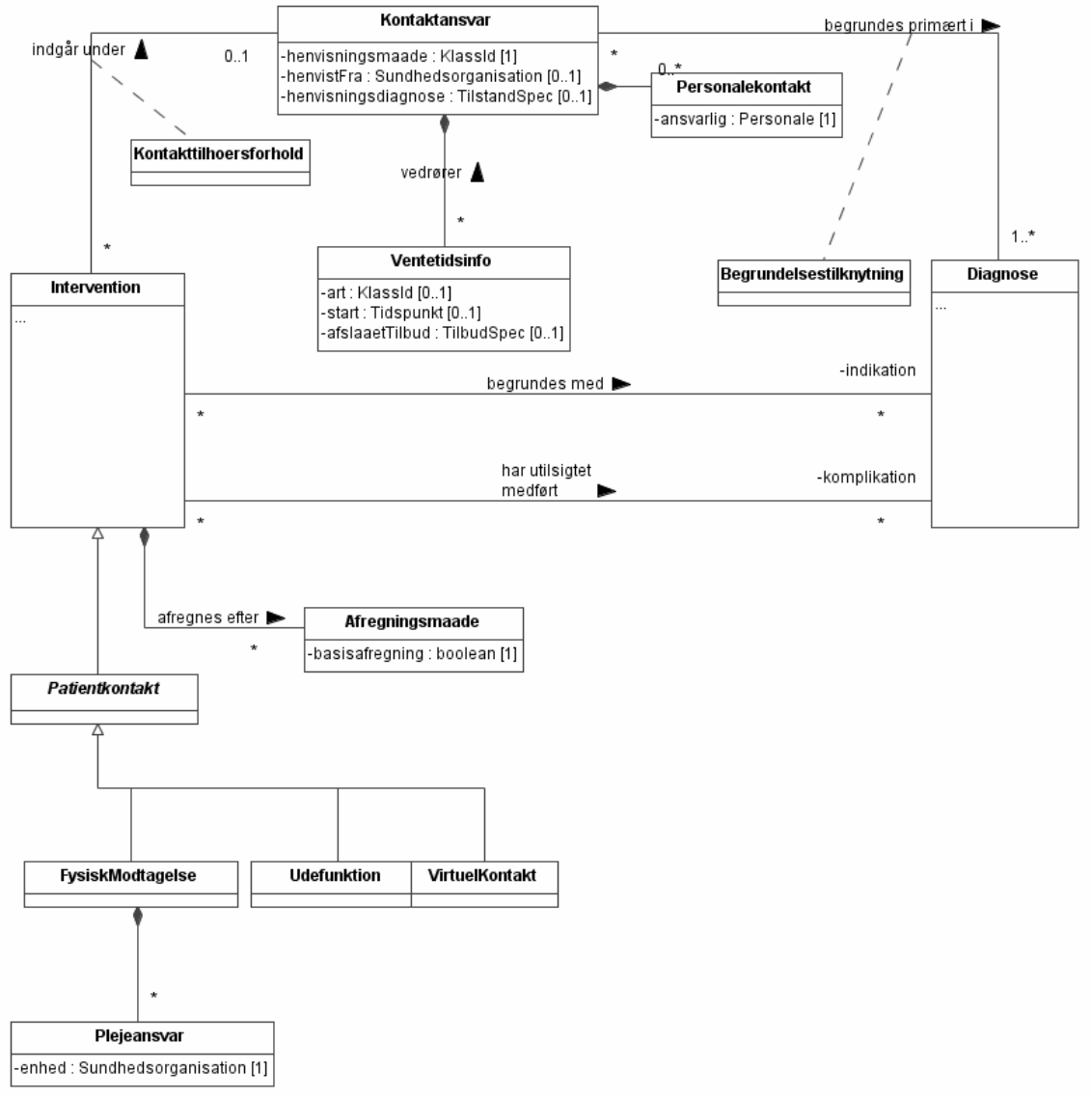


Figure 1 ”Patient encounter” class diagram in GEPJ 2.2

The definitions of the classes are [1] (translated):

Kontaktansvar (EncounterResponsibility): Specifies the responsibility relationship between an organizational unit and one or more patient encounters.

Patientkontakt (Patient Encounter): Unites the interventions, which consist of contact between health services and patient – e.g. inpatient, outpatient visit, telephone consultation. Contact responsibility is a requirement for the existence of a patient encounter. This also applies for those parts of the health service that are not protected by a gate-keeper function – e.g. emergency room, emergency doctor.

FysiskModtagelse (PhysicalReception): Expresses a specific encounter between health services and patient where the patient is physically present at the healthcare provider – e.g. inpatient, outpatient. The attribute udfoerendeEnhed (PerformingUnit) on associated Status-objects describe the unit where the patient is received.

Udefunktion (OutFunction): Expresses specific contact between health services and patient, where representatives for the health services – for the sake of a specific patient – are physically present outside the provider premises – e.g. home visit, employer visit.

VirtuelKontakt (VirtualContact): Expresses virtual – i.e. non-physical – encounter between health services and patient – e.g. telephone consultation, email.

Plejeansvar (CareResponsibility): The unit that, at a specific time, has care responsibility. A closed CareResponsibility means: Responsibility has ended.

Ventetidsinfo (WaitingTimeInfo): Waiting time may be separated into periods. The start of these periods is defined by ‘start’ and runs until the ‘start’ of the following WaitingStatus.

Personalekontakt (ContactProfessional): Contact person for a certain responsibility. A closed ContactProfessional means: Contact responsibility has ended.

Afregningsmåde (ReimbursementMethod): INFORMATIVE. Reimbursement is not a part of GEPJ, and thus it is not mandatory to implement this class.
The two association classes are defined by:

Kontakttilhoersforhold (EncounterBelongingTo): No definition in GEPJ 2.2

Begrundelsestilknytning (JustificationAssociation): Association class: Connects a ContactResponsibility to a diagnosis. States that the connected diagnosis is the main reason for creation of the ContactResponsibility.”

6.3 HL7 Patient Administration

Patient administration in HL7 is partly described in **Fejl! Henvisningskilde ikke fundet.**:, as far as the D-MIM is concerned, while specific R-MIMs, triggers etc. regarding encounters are described in [4]:.
The total D-MIM for patient administration is quite complex. The following is an extract, concentrated on the actual encounter.

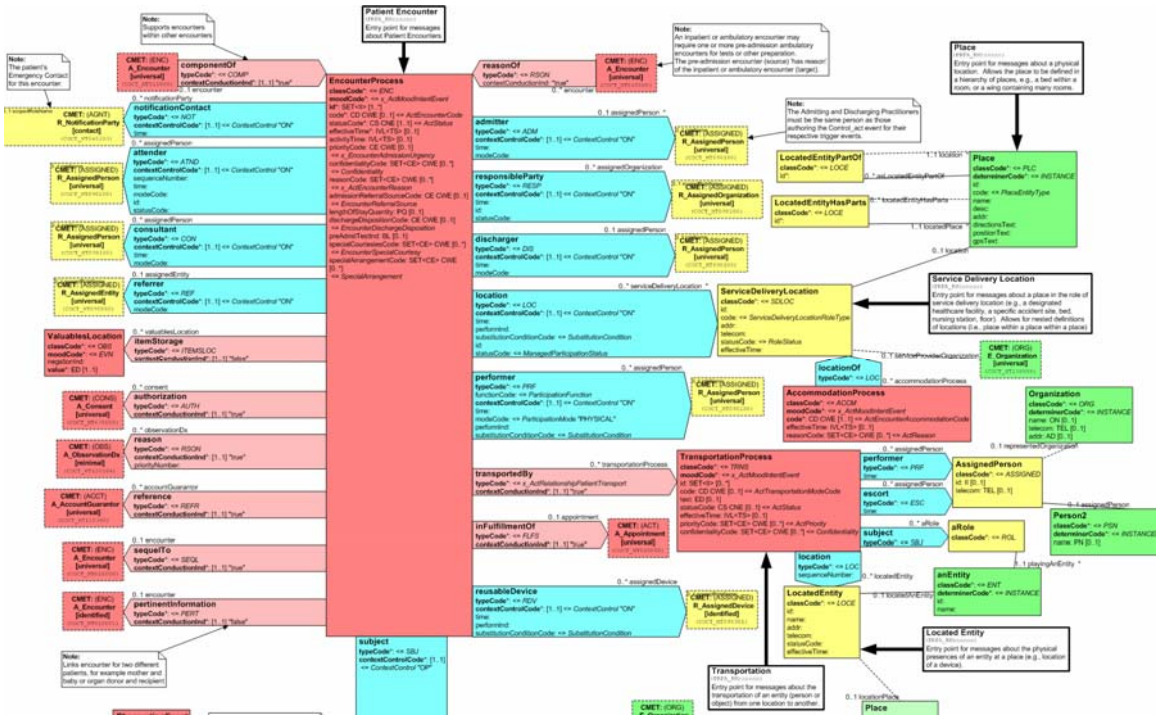


Figure 2 Extract of the Patient Administration D-MIM in HL7

The core elements in the diagram are:

Encounter Process - An interaction between a patient and care provider(s) for the purpose of providing healthcare-related service(s). This can be a request, appointment, or event.

componentOf - An optional association from the focal encounter (target) to a super-encounter (source) of which it is a component. Examples include: (1) clinical encounters with radiology and clinical specialists within a single outpatient visit, (2) clinical encounters with an eye specialist for blurry vision and an orthopedist for a sprained knee within an emergency room visit. The contextConductionInd is set to "true" because both encounters have the same subject (patient).

reference (AccountGuarantor) - An optional association from the focal encounter (source) to one or more patient accounts (target). The *contextConductionInd* is set to "true" because the encounter and the account have the same subject (the patient). Each account with its associated account holder (guarantor) is sent in a separate A_AccountGuarantor CMET.

reason (ObservationDx) - An optional association from the focal encounter (source) to one or more diagnoses (target). The *contextConductionInd* is set to "true" because the encounter and the observation have the same subject (the patient). Each diagnosis is sent in a separate *A_ObservationDx* CMET. The type of diagnosis is determined by the *Observation.code* within the *A_ObservationDx* CMET; "ADMX" for an admission diagnosis, "INTDX" for an interim diagnosis, and "DISDX" for a discharge diagnosis. Multiple diagnoses of the same type can be rank ordered using the *reason.priorityNumber*.

responsibleParty - Optional association to one or more healthcare provider organizations that hold clinical responsibility for the patient encounter. *Note: only one responsibleParty participation should be in the active state at a time.*

consultant - Optional association to advisor(s) participating in this patient encounter by performing evaluations and making recommendations

attender - Optional association to the healthcare provider who has responsibility for overseeing a patient's care during a patient encounter. *Note: only one attender would be in the active state at a time.*

6.4 ContactResponsibility and associations

Besides the exact definition in the UML report within GEPJ the intention behind the EncounterResponsibility is also described in [2]:

“An encounter responsibility states that – within is duration– it is a responsibility relationship between a certain unit (the responsible unit) and a patient regarding a certain health condition. This responsibility implies, that the unit accepts to keep contact to the patient about the problem that is the reason for the encounter responsibility, that is to call in the patient to outpatient visits, inpatient encounters etc. and in connection to that, plan and execute the pure clinical episode of care in the form of examinations, treatment, rehabilitation etc, that has to be done at and between the specific patient encounters.”

The intention may thus be expressed as

- Being able to connect related patient encounters to one coordinated episode of care

- Being able to connect one single responsible unit to this episode of care

This is modelled in HL7 through the usage of EncounterProcess to describe both the individual patient encounters and to describe the overall episode of care.

The separation of ContactResponsibility and PatientEncounter is found through the EncounterProcess.code attribute.

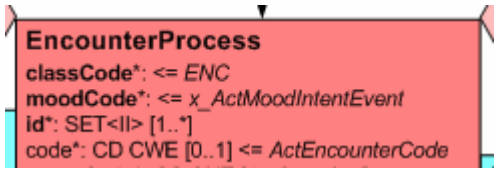


Figure 3 EncounterProcess - code

HL7 standard vocabulary for ActCode does not contain a suitable code to describe the ContactResponsibility type. Act.code is however a CWE (coded with extensions), and so it is possible to specify a realm-specific code list with the relevant value. As a consequence, it is necessary to use a realm-specific code list for Denmark until a suitable code is included in the standard HL7 vocabulary.

6.5 Attributes

Henvisningsdiagnose (ReferralDiagnosis): “Specifies the diagnosis, that was the reason for the creation of the EncounterResponsiblity, if the patient is referred from another party.“

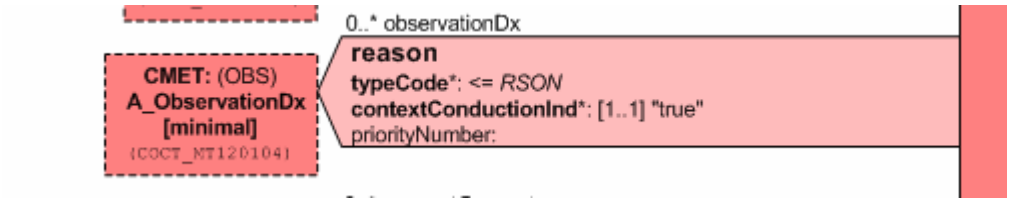


Figure 4 ReferralDiagnosis in HL7

This is specified in HL7 using the reason act relationship and the corresponding *A_ObservationDx* CMET. Observation.code is "ADMX" (admission diagnosis).

Henvisningsmaade (ReferralSource). “Specifies the source of the referral, e.g. no referral, born here, abroad, GP, hospital unit. “

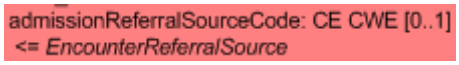


Figure 5 Referral source in HL7

This is the admissionReferralSourceCode attribute on Encounter in HL7. A national Danish coding scheme is necessary, as the HL7 standard code list is insufficient.

HenvistFra (ReferredFrom): ”Specifies the organizational unit that referred the patient – if relevant”.

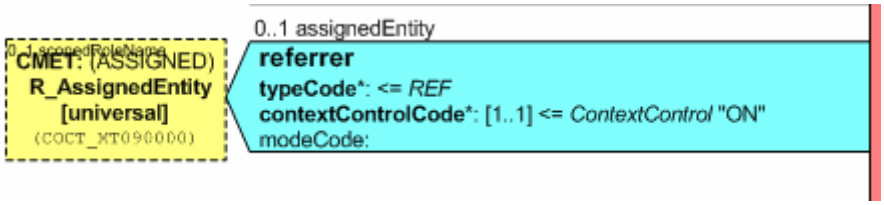


Figure 6 Referrer in HL7

This is the referrer (R_AssignedEntity) in HL7.

6.6 WaitingTimeInfo

WaitingTimeInfo does not have a corresponding concept within the HL7 patient administration topic.

The intention behind WaitingTimeInfo is to be able to monitor and react on waiting times for patients in different phases of the episode of care. In [5] this is described by the means of a timeline with a number of different events.

When representing the intention in HL7 one may chose to establish a new specific model to calculate waiting times. In this case the individual waiting time is a ‘waiting calculation act’ (in much the same way as an invoice line is a calculation based on a complex clinical process). Another representation may be to use the individual events in the clinical/administrative process. This latter solution an algorithm based on the events will be the calculation of the different waiting times.

A specific modelling will have to be determined by the surrounding factors of the actual scenario.

6.7 ContactProfessional

ContactProfessional in HL7 Patient Administration are two different participations on the instance of EncounterProcess, which represents the ContactResponsiblity.

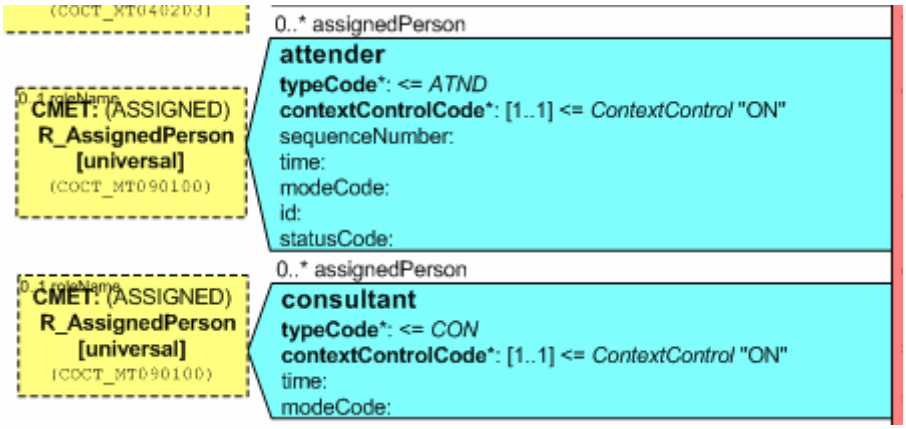


Figure 7 ContactProfessional in HL7

HL7 distinguishes two different contacts between the patient and the professionals: either as responsible or as advisor for the responsible. In GEPJ this distinction is not made.

6.8 Attributes

Ansvarlig (Responsible): No definition in GEPJ.
Responsible has the data type Personale (Professional), which has the attributes personaleId (employeeId), rolle (role), signature (signature), tilknyttetEnhed (associatedUnit).

The individual attributes are found in R_AssignedPerson.id (employeeId), R_assignedPerson.code (role), R_assignedPerson.certificateText (signature) and representedOrganisation (associatedUnit).

6.9 JustificationAssociation

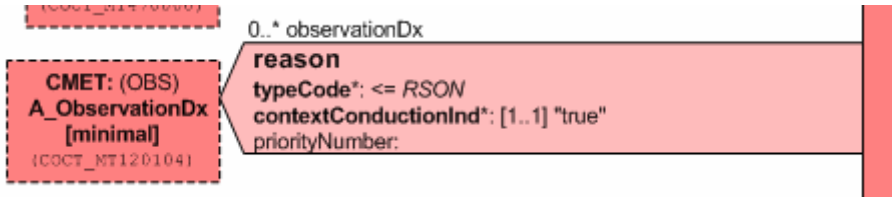


Figure 8 JustificationAssociation in HL7

JustificationAssociation is specified in HL7 by using reason (ObservationDx) act-relationship from the EncounterProcess, which represents the ContactResponsibility. Cardinality has to be restricted to 1.*. Observation.code is "INTDX" (intermediate diagnosis) or ”DISDX” (discharge diagnosis).

6.10 EncounterBelongingTo

The general association between intervention and ContactResponsibility is not covered by the HL7 patient administration topic. Instead it is modelled in the domains where it is

considered meaningful to relate act with an encounter. A more restricted relation between Encounter and ContactResponsibility is modelled using the componentOf act-relationship.

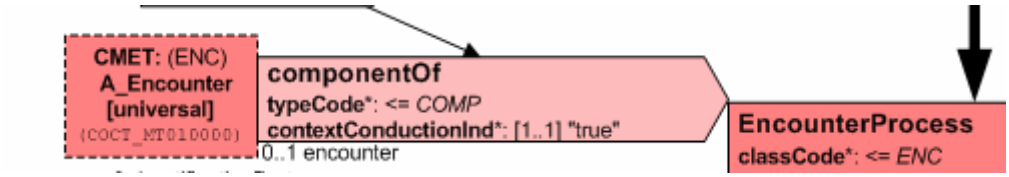


Figure 9 Relation between PatientEncounter and ContactResponsibility

6.11 PatientEncounter, specialisations and relations

PatientEncounter is represented directly by EncounterProcess in HL7.

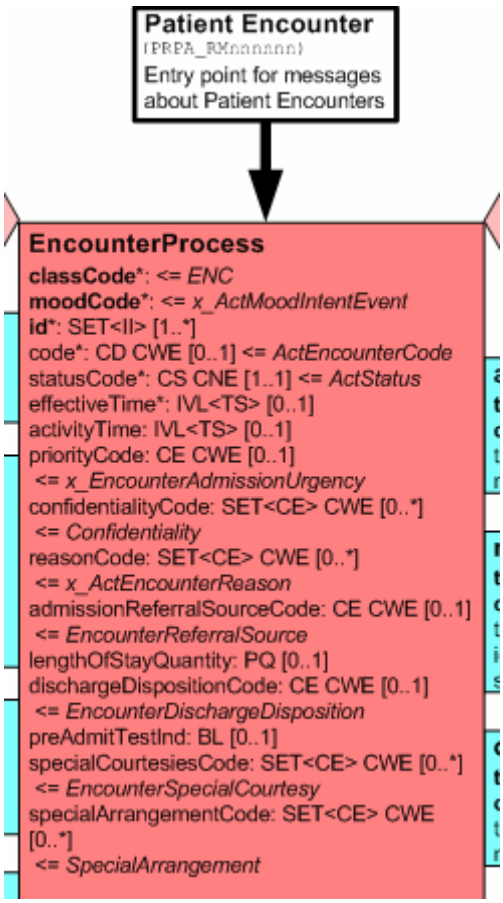


Figure 10 PatientEncounter in HL7

6.12 PhysicalReception

PhysicalReception is expressed in HL7 by a number of different types of encounters. In HL7 standard vocabulary PhysicalReception corresponds to an encounter with EncounterProcess.code ’IMP’ (Inpatient), ’AMB’ (ambulatory), ’EMER’ (emergency) or ’SS’ (Short Stay).

6.13 OutFunction

OutFunction is expressed by an EncounterProcess with EncounterProcess.code 'FLD' (field) from HL7 standard Vocabulary.

6.14 VirtualEncounter

The class is expressed by an EncounterProcess with EncounterProcess.code'VR' (virtual) from HL7 standard vocabulary.

6.15 CareResponsibility

CareResponsibility is expressed in HL7 through responsibleParty (R_AssignedOrganization). To be able to distinguish between other types of responsibility AssignedOrganisation.code may be assigned 'NCCF' (Nursing or custodial care facility).

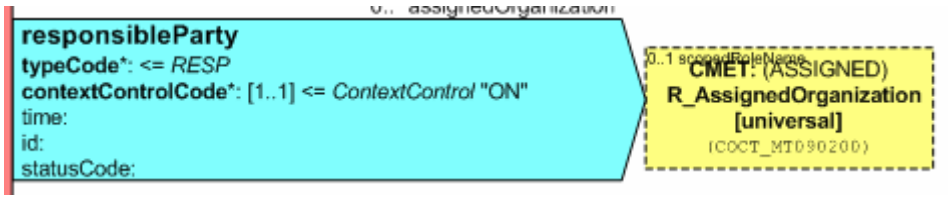


Figure 11 Plejeansvar in HL7

Attributes
Enhed (Unit): Is represented in AssignedOrganization.code.

6.16 ReimbursementMethod

ReimbursementMethod is expressed by the usage of reference (A_AccountGuarantor) act-relationship.

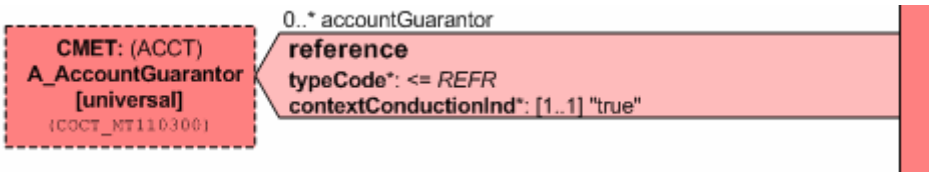


Figure 12 ReimbursementMethod in HL7

6.17 Attributes

Basisafregning (BasicReimbursement): "A 'no' value specifies that charged reimbursement is to be used. A 'yes' value specifies basis reimbursement".

Is expressed by Account.id – as basis reimbursement is modelled as an account. GuarantoRole.id is assigned a value, which identifies the responsible for the common reimbursement (e.g. National Board of Health).

6.18 Conclusion

It is possible to represent all concepts and associations from the GEPJ patient encounter part to the HL7 standard model for patient administration. The only exception is WaitingTimeInfo, which does not have a corresponding concept in HL7. The concept WaitingTimeInfo is considered of no clinical importance.

The HL7 Patient administration model is significantly more comprehensive than the model in GEPJ.

7 Data types

7.1 Introduction

In this chapter the data types in HL7 and GEPJ are compared. The definitions and documentation of the data types in the two standards are compared. The fundamental data types are identified. Finally the data types defined in the two standards are compared by representing GEPJ data types in terms of HL7 data types.

When comparing data types in HL7 and in GEPJ it is revealed, that the use of data types are fundamentally different:

- In HL7 a data type represents a type of value. An HL7 data type can be used to describe only one property of an entity. Data types in GEPJ can describe more properties of an entity - e.g. a drug or an employee.
- In HL7 data type attributes express different aspects of the data type itself; attributes in GEPJ can express an axis of the data value – or even different, related values.
- HL7 uses data types for specific kinds of data related to a specific kind of entity (like e.g. Person name, address etc.). This kind of data types are not used in GEPJ.

Thus both the use of data types and the use of data type attributes differ. This will however, not necessarily generate problems in a mapping from GEPJ into HL7, as the value of one attribute in GEPJ can be mapped into more attributes in HL7.

In section 4.3 it is described, that data types are documented differently in HL7 and GEPJ:

- Data types are described quite thoroughly in HL7. Compared to the HL7 description, only the ‘Primary properties’ and parts of the ‘Declaration’ are given in the GEPJ documentation. It makes the documentation easier to read, but probably less exact.

  It should, be considered whether there is a need for at more exact description of the GEPJ data types. For that purpose the HL7 form – or parts of it - could be used. E.g. it should be considered to use the HL7 way of specifying different specializations of the generic data types. Instead of stating restrictions to ‘KlassId’ textually, a formal description, like in HL7 should be considered.

- Data types are documented textually in HL7. In GEPJ it is based in an UML diagram. GEPJ’s use of the UML representation of the data types does not add substantially to the understanding. There are no relations from the rest of the GEPJ model to the model of data types. In the rest of the GEPJ model data types are used as types of attributes; not really as objects.

As will be shown in section 4.4, data types in the two standards are constructed differently:

- Basic data types like integer and string are redefined within HL7. They are not in GEPJ.

The definitions of basics in HL7 can be hard to read. Instead of giving a clear understanding these descriptions tend to make you doubt the understanding you had already. To support the reader, and make the documentation more intelligible, there is a textual description of the HL7 data types. The textual description may be quite cryptic too though (e.g.: 'Integer numbers are precise numbers…' It is not described what is meant with 'precise').

- Both GEPJ and HL7 use composite data types based on basic types and other composite data types. Most composite data types in GEPJ and HL7 are different. GEPJ data types are typically mapped into the generic data types in HL7.

7.2 Definition of Data Types

In HL7 data types are defined like this:

> Data types define the meaning (semantics) of data values that can be assigned to a data element.
>
> Data type values stand for themselves, the value is all that counts, neither identity nor state or changing of state is defined for a data value.
>
> One can think of data values as immutable objects where identity does not matter (identity and equality are the same.)

Data types are not explicitly defined in GEPJ. Thus GEPJ is based on a 'common' understanding of data types, which corresponds to the definition given in HL7.

7.3 Documentation of Data Types

In the following it is described how data types are described in HL7 and GEPJ. Based on the comparison, it can be considered whether it is useful to describe GEPJ data types in the same way as HL7 data types.

7.3.1 Documentation of Data Types in HL7

In HL7 a data type is described by the following information:

- Primary properties
- Declaration
- Invariant Statements

Primary Properties
At the beginning of many data types the "primary" properties are given (e.g. Name, Type, and Definition)

The definition of INT is e.g.:

> Integer numbers (-1,0,1,2, 100, 3398129, etc.) are precise numbers that are results of counting and enumerating. Integer numbers are discrete, the set of integers is infinite but countable. No arbitrary limit is imposed on the range of integer numbers. Two NULL flavors are defined for the positive and negative infinity.

Declaration
Every data type is declared in a form that begins with the keyword **type**.

The declaration of INT is e.g.:

```
type IntegerNumber alias INT specializes QTY {
        INT  successor;
        INT  times(INT x);
        INT  predecessor;
        INT  negated;
        BL   isNegative;
        BL   nonNegative;
        INT  dividedBy(INT x);
        INT  remainder(INT x);
        BL   isOne;
  literal  ST;
};
```

If any, attributes will be described at the top of the type declaration.

Invariant Statements

Invariant statements are logical statements that are true at all times. The invariant statements are considered to be 'the true definition' of the data type.

One of the invariant statements for INT is e.g.:

```
invariant(INT x, o, i) where x.nonNull.and(o.isZero)
{
   x.lessThan(x.successor);
   x.plus(o).equal(x);
   x.plus(y.successor).equal(x.plus(y).successor);
   x.times(o).equal(o);
   x.times(y.successor).equal(x.times(y)).plus(x);
};
```

Literal Form

A literal is a character string representation of a data value. Literals are defined for many types. A literal is a type conversion from and to a Character String (ST) with a specially defined syntax.

The literal form of INT is e.g.:

```
INT.literal ST {
   INT digit : "0"        { $.isZero; }
             | "1"        { $.equal(0.successor); }
             | "2"        { $.equal(1.successor); }
             | "3"        { $.equal(2.successor); }
             | "4"        { $.equal(3.successor); }
             | "5"        { $.equal(4.successor); }
             | "6"        { $.equal(5.successor); }
             | "7"        { $.equal(6.successor); }
             | "8"        { $.equal(7.successor); }
             | "9"        { $.equal(8.successor); };

  INT uint   : digit      { $.equal($1); }
             | uint digit  { $.equal($1.times(9.successor).plus($2)); };

  INT        : uint       { $.equal($1); }
             | "+" uint    { $.equal($2); }
             | "-" uint    { $.equal($2.negated); };
};
```

7.3.2 Documentation of data types in GEPJ

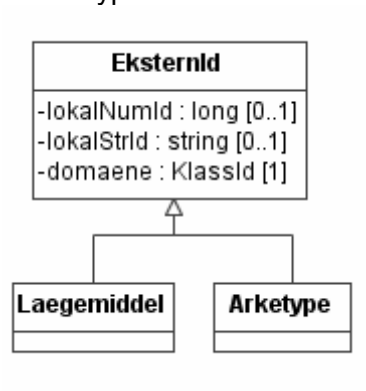
Data types are documented using UML diagrams in GEPJ. The diagrams are supplemented by:

- A general description,
- A list of attributes and
- A list of relations in the UML representation.

Basic data types like string and integer are not included in the UML diagram but listed with the other data types.

In the following the documentation of the data type EksternId (specifies a globally unique identification of a non GEPJ object).

UML Diagram
The UML representation of the data type EksternId:



General Description
The general data type description for the data type EksternId:

General	
Name	EksternId
Visibility	public
Active	false
Abstract	false
Leaf	false
Root	false
Owner	Data types

This part of the description is the only part of the basic data types.

Attributes
A list of attributes is given for composite data types. For EksternId:

Attributs		
Name	**Type**	**Initial Value**
domaene	KlassId	
lokalNumId	long	
lokalStrId	string	

Relations
A list of relations to the data type in the UML representation is given. For EksternId:

Relations			
Name	**Type**	**Begins**	**Ends**
<unnamed>	generalization	Laegemiddel	EksternId
<unnamed>	generalization	Arketype	EksternId

7.4 Fundamentals in Data Types

In this section we make an overall comparison, by comparing the building blocs from which the data types are constructed:

- The types of data types
- The generic data types – prepared for the user defined data sets and
- The abstract data types.

The intention is to point out similarities and differences.

The Types of Data Types
Data Types in HL7 are divided into:

- Basic types,
- Generic Collections,
- Generic Type Extensions and
- Timing Specification.

In GEPJ there are:

- Basic types and
- Constructed types.

Composite Ddata Ttypes
In GEPJ the following basic data types are used:

- boolean
- datetime
- float
- int
- long
- string

The rest of the GEPJ data types are build from these basic data types. The basic data types are considered well known and are not redefined in the context of GEPJ. Some composite data types are used to describe values for a specific kind of entity: e.g. AktivitetSpec (clinical act specification), Anatomi (describing the anatomy of a person), Laegemiddel (describing a medicine) etc. Attributes in these composite data types are used to express different properties about this entity.

HL7 has a large number of 'Basic Types'. The base for all these data types is the abstract type DataValue (ANY). All other data types are specializations.

HL7 does contain what is considered the basic types in GEPJ, but they are not described differently than the more complex or special-purposed data types.

The basic types defined for GEPJ are specialized like this in HL7:

- Boolean (**BL**) specializes DataValue (**ANY**)
- Point in time (**TS**) specializes Abstract Type Quantity (**QTY**) specializes DataValue (**ANY**)
- Real Number (**REAL**) specializes Abstract Type Quantity (**QTY**) specializes DataValue (**ANY**)
- Integer Number (**INT**) specializes Abstract Type Quantity (**QTY**) specializes DataValue (**ANY**)
- Character String (**ST**) specializes Encapsulated Data (**ED**) specializes Binary Data (BIN) specializes LIST< BooleanNonNull (**BN**) specializes Boolean (**BL**) specializes DataValue (**ANY**)>

The composite data types in GEPJ can be represented in terms of the HL7 data types (see chapter 5.7). Data types are typically used to describe values of a specific type of data (not a specific type of entity). Thus attributes in composite data types describe different properties about the data value. Data types like Telecommunication a list of Address Part), Postal Address and Person Name, are used to describe one property about a specific kind of entity. They do not represent a set of values.

Generic Collections
In HL7 a number of data types that can "collect" other data values are defined:

- Set,
- Sequence,
- Bag and
- Interval

There are no corresponding data types defined for GEPJ. The multiplicity of an attribute is defined for the attribute itself, not the data type. E.g. Paaroerende.postadresse (address of a relative) is of type string and has the multiplicity *. Person.add has the type BAG<AD> (still with the multiplicity [0..*] though).

Thus GEPJ has not the strength to describe differences between bags, lists and intervals using data types.

Generic Type Extensions

In HL7 generic type extensions are 'generic types with one parameter type, and that extend (specialize) their parameter type'. They give the option to extend any other data type with information about data history or probability/certainty of a value. It seams, that the generic type extensions still is not in use.

There is nothing like that in GEPJ data types. History is modeled within the GEPJ data model and uncertainty is not generally supported. The certainty attribute could probably be valuable when describing a diagnosis.

Comparison:

The concept of basic data types is different in GEPJ and HL7. The basic data types in GEPJ are included in HL7, but are redefined within HL7. The HL7 redefinition seems to be equivalent to the common understanding of the basic data types. Thus no conflicting definitions are identified.

Both standards use data types constructed and specialized from other data types.

HL7 contains generic collections and generic type extensions, which are not defined for GEPJ. Thus the ability for defining complex data types in GEPJ is less available than in HL7.

Generic data types

Some of the data which is included in an HL7 message is classified in the messaging system. This can e.g. be a diagnosis or a procedure. There's a need for data types which can be customized to individual or local use of the standard.

There is one data type in GEPJ open to local classifications:

class KlassId; Can contain a concept as a part of a coded range.

A number of other data types have attributes with this data type.

KlassId may be explicitly restricted to a specific classification in the standard. This will be mentioned in the textual description of the attribute (e.g. Person.identifikaton, MaalSpec.operator).

HL7 has a number of data types, which makes room for local classifications:

Coded Value (CV) specializes *CE (*specializes *CD)*
Definition:
Coded data, specifying only a code, code system, and optionally display name and original text. Used only as the type of properties of other data types.

and

Concept Descriptor (CD) specializes *ANY*
Definition:
A *CD* represents any kind of concept usually by giving a code defined in a code system. A *CD* can contain the original text or phrase that served as the basis of the coding and one or more translations into different coding systems. A *CD* can also contain qualifiers to describe, e.g., the concept of a "left foot" as a postcoordinated term built from the primary code "FOOT" and the qualifier "LEFT". In cases of an exceptional value, the *CD* need not contain a code but only the original text describing that concept.

Some code domains are qualified such that they include the portion of any pertinent local coding system that does not simply paraphrase the standard coding system (*coded with extensibility*, CWE.) If a CWE qualified field actually contains such a local code, the coding system must specify the local coding system from which the local code was taken. However, for CWE domains the local code is a valid member of the domain, so that local codes in CWE domains constitute neither an error nor an exceptional (NULL/other) value in the sense of this specification.

Through the refinement process the CD may be restricted.

Example of use 1:
In the Reference Information Model (RIM):
Act.code: CD (Vocabulary domain: ActCode (CWE))

In the Care Provision Domain Message Information Model (D-MIM)
CareProvision.code: CD CWE <= ActCode

As the vocabulary domain associated with a particular attribute instance is designated by the <= the vocabulary of CareProvision.code is ActCode.

Care Record Refined Message Information Model (R-MIM) for the occurrence of the care provision event:
CareProvisionEvent.code: CD CWE[0..1] <= ActEncounterCode;

The data set of ActEncounterCode contains a fairly restricted set of medical services. But as the *coding strength* of CD is CWE the end user is allowed to send a term from an arbitrary coding system in place of the term from the specified value-set.

Example of use 2:
In the Clinical Statement R-MIM:
Observation.code: CD CWE [0..1] <= ObservationType

ObservationType '*Identifies the kinds of observations that can be performed*'.

There is an implementation guide on how to use SNOMED CT® in HL7. This guideline describes how to e.g. use SNOMED CT® for the Oberservation.code. There is no explicit reference to SNOMED CT® in the R-MIM.

Comparison:

Both standards make room for local classifications. Because of the refinement and localization HL7 has a hierarchy of generic data types, which may be stricter in the R-MIMs.

Abstract Data Types
In this section it is investigated if both standards have the need for abstract data types.

In HL7 two abstract data types are defined:

- ANY; abstraction of all data types and
- QTY

Abstract Type Quantity (QTY) specializes **ANY** **Definition:** The quantity data type is an abstract generalization for all data types (1) whose value set has an order relation (less-or-equal) and (2) where difference is defined in all of the data type's totally ordered value subsets. The quantity type abstraction is needed in defining certain other types, such as the interval and the probability distribution.

QTY generalizes:
- Integer Number (INT)
- Real Number (REAL)
- Ratio (RTO)
- Physical Quantity (PQ)
- Money Amount (MO)
- Point in time (TS)

Physical Quantity is used for measurement result and contains information on unit of measurement.

GEPJ has the abstract data types:
- NyfoedtSpec which generalizes stillborn and new born. Used for reporting only.
- VaerdiType which generalizes a number of different types of values.

VaerdiType generalizes:
- VaerdiDectal; value of type float; unit of measure are given.
- VaerdiHeltal; value of type integer; unit of measure are given.
- VaerdiKlass; value of type KlassId
- VaerdiEkstern; value of type EksternId (external identification).
- VaerdiStreng, Value of type string.
- VaerdiBool; value of type boolean.

Thus VaerdiType corresponds almost to the abstract HL7 data type QTY.

Comparison:

Abstract data types are used in both standards. One of the data types (**VaerdiType** and **Abstract Type Quantity**) is quite similar.

7.5 Mapping data types

In order to verify, that GEPJ information can be formed into HL7 messages, it does not make sense just to map GEPJ data types into HL7 data types. When mapping data represented in GEPJ into an HL7 message, you will have to map each of the contained data elements. Data elements represented in GEPJ will, of course, be typified by the GEPJ data types. Likewise, the data elements in HL7 are typified by HL7 data types. Given any mapping from a GEPJ data element to a corresponding HL7 data element, we cannot be sure the two types of data elements are typified by what we will consider

to be equivalent data types. Thus being able to map all GEPJ data types into HL7 data terms does not really prove anything. *Not* being able to map GEPJ data types into HL7 data types will however, prove that GEPJ cannot be mapped into HL7. Therefore this document includes a description of how to define the GEPJ data types in terms of the HL7 data types.

In the following table all the GEPJ data types are listed. Each of the data types are expressed in the terms of a HL7 data type. The GEPJ data type names and attribute names are given in Danish. Data types of attributes are given in parentheses after the attribute. A definition is given for a HL7 data type, when it is referred to for the first time in the table.

Some of the GEPJ data types are introduced in order to maintain the existing reporting to the national register of persons. Thus they will presumably be excluded from the GEPJ data types when the reporting is changed. The data types are given in a gray color. Mapping data does not require mapping of abstract data types. Therefore the abstract data types are omitted from the table.

Name	Description	Attributes	HL7 data type	Definition	Comments
AktivitetSpec	Specifies an ”aktivitet” (a clinical act)	benaevnelse (KlassId) lokalisation (Anatomi) morfologi (KlassId) observationsemne (KlassId) proevemateriale (KlassId)	Concept Descriptor (CD) if represented as one multi axial value set. Alternatively SET<CD>	A *CD* represents any kind of concept usually by giving a code defined in a code system. A *CD* can contain the original text or phrase that served as the basis of the coding and one or more translations into different coding systems. A *CD* can also contain qualifiers to describe, e.g., the concept "left foot" as a post coordinated term built from the primary code "FOOT" and the qualifier "LEFT". In cases of an exceptional value, the *CD* need not contain a code but only the original text describing that concept	An ”aktivitet” will be implemented as an Act in HL7. The type of Act is CD. The attributes of CD are: code (ST) codeSystem (UID) codeSystemName (ST) codeSystemVersion (ST) displayName (ST) originalText (ED) translation (SET<CD>) qualifier (LIST<CR>)
Anatomi	Specifies an anatomical localization.	fokus (KlassId) lateralitet (KlassId)	Concept Descriptor (CD) if represented as one multi axial value set. Alternatively SET<CD>		The HL7 attribute “Observation.targetSiteCode” is a set of CDs
Arketype	Specifies an archetype – e.g. *blood pressure*.		Concept Descriptor (CD)		

Name	Description	Attributes	HL7 data type	Definition	Comments
DoedfoedtSpec	Contains information about a stillborn.	skoennetDoedstidspunkt (Tidspunkt) + the attributes inherited from the abstract data type NyfoedtSpec (specifies a new born).	All values can either be encapsulated in one Concept Descriptor (CD) or each of the attributes can be given a data type. The attributes all have data types like INT, REAL, CS or TS.		LivingSubject.deceasedTime:: TS (0..1)
EksternId	Specifies a globally unique identification of a non GEPJ object – e.g. an x-ray in a data bank.	domaene (KlassId) lokalNumId (long) lokalStrId (string)	Instance Identifier (II)	An identifier that uniquely identifies a thing or object. Examples are object identifier for HL7 RIM objects, medical record number, order id, service catalog item id, Vehicle Identification Number (VIN), etc. Instance identifiers are defined based on ISO object identifiers.	The attributes of II are: • root (Code System(UID)) ~ could be mapped from domaene.lokalStrId • extension (ST) ~ could be mapped from lokalNumId • assigning Authority Name (ST) – not used in GEPJ • displayable (BL) – not used in GEPJ
KlassId	Specifies a	replikeret	Concept		See e.g. the

Name	Description	Attributes	HL7 data type	Definition	Comments
	concept which is a part of a coded range.	(Tidspunkt) termId (long)	Descriptor (CD)		section Generic data types
Laegemiddel	Specifies a unit of a drug – e.g. “resoriblet Nitromex 0,25 mg”.		Coded With Equivalents (CE)	Coded data that consists of a coded value and, optionally, coded value(s) from other coding systems that identify the same concept. Used when alternative codes may exist.	Medicin.code (CE CWE => x_Medicin (that is the local classification of drugIds))
LevendefoedtSpec	Specifies a living newborn.	apgar5 (VaerdiHeltal) + the attributes inherited from the abstract data type NyfoedtSpec (specifies a new born).	All values can either be encapsulated in one Concept Descriptor (CD) or each of the attributes can be given a data type. The attributes all have data types like INT, REAL, CS or TS.		“An APGAR score would be implemented as an AssessmentScaleEvent observation.” AssesmentScaleEvent.value (INT)
MaalSpec	Specifies a property of a result	operator (KlassId) ref1 (VaerdiType) ref2 (VaerdiType) sti (string)	Concept Descriptor (CD) or SET<CD>		
MisdannelseSpec	Describes malformations of a newborn.	art (KlassId) misdannelserUndersoegt (KlassId)	Concept Descriptor (CD)		

Name	Description	Attributes	HL7 data type	Definition	Comments
Personale	Specifies an employee – e.g. doctor P. Andersen.	personaleId (EksternId) rolle (KlassId) signatur (string) tilknyttetEnhed (Sundhedsorganisation)	Types of attributes to Employee		An employee in **Personnel Management Domain Information Model** has the following attributes and relations: Employee.id (SET<II>) ~ could be mapped from personaleId Employee.jobCode (CE) ~ could be mapped from rolle Employee-> PrincipalPerson.name (BAG<EN>) ~ could be mapped from signature Employee-> employerOrganization -> Organization.id (SET<II>) ~ could be mapped from tilknyttetEnhed
Stadium	Specifies a state of a health condition.	skala (KlassId) skalavaerdi (VaerdiType)	Concept Descriptor (CD)		Did not find a corresponding attribute in HL7. Presumably the information can be part of the condition code.
Sundhedsorganisation	Specifies an organizational unit – e.g. Department M at a specific hospital.		Organization Name (ON)	An EN used when the named Entity is an Organization. A sequence of name parts.	
Tidsinterval	Specifies a time period – e.g. 2	sekunder (long)	Abstract Type	Abstract Type Quantity (QTY)	

Name	Description	Attributes	HL7 data type	Definition	Comments
	days, 5 hours and 23 seconds.		Quantity (QTY)	of time	
Tidsperiode	Specifies a specific periode of time – e.g. from 08:51:12 at the 18th of April 2003 to 23:13:00 at the 20th of April 2003.	fra (Tidspunkt) til (Tidspunkt)	Interval (IVL<Time>)	A set of consecutive values of an ordered base data type.	
Tidspunkt	Specifies a point in time – e.g. 08:46:16 at the 18th of April 2003.		Point in Time (TS)	A quantity specifying a point on the axis of natural time. A point in time is most often represented as a calendar expression.	
TilbudSpec	Contains information that a patient has been offered treatment by some other health organization.	datoTilbud (Tidspunkt) tilbudtAfdeling (Sundhedsorganisation)	Concept Descriptor (CD)		
TilstandSpec	Specifies af health condition.	aktualitet (KlassId) benaevnelse (KlassId) lokalisation (Anatomi) morfologi (KlassId) patologiskProces (KlassId) svaerhedsgrad (KlassId) sygdomsudvikling (KlassId) tilstandstadium (Stadium) udloesendeAge	Concept Descriptor (CD)		

Name	Description	Attributes	HL7 data type	Definition	Comments
		ns (KlassId)			
VaerdiBool	Can be either SAND (true) or FALSK (false). Specializes VaerdiType.	enhed (Klassid) = NULL vaerdi (boolean)	Boolean (BL)	*BL* stands for the values of two-valued logic. A *BL* value can be either *true* or *false*, or, as any other value it may be NULL.	
VaerdiDectal	Specifies value and unit of value – e.g. 45,7 kg or 0,27 mmol. Specializes VaerdiType.	enhed (Klassid) vaerdi (float)	Physical Quantity (PQ)	A dimensioned quantity expressing the result of measuring.	
VaerdiEkstern	Can contain a global unique identification of a non GEPJ object – e.g. an x-ray in a data bank. Specializes VaerdiType.	enhed (Klassid) = NULL vaerdi (EksternId)	Instance Identifier (II)		
VaerdiHeltal	Specifies value and unit of value – e.g. 4 ml/min. Specializes VaerdiType.	enhed (Klassid) vaerdi (integer)	Physical Quantity (PQ)		
VaerdiKlass	Can contain a specification of a concept which is a part of a coded range. Specializes VaerdiType.	enhed (Klassid) = NULL vaerdi (KlassId)	Concept Descriptor (CD)		
VaerdiStreng	Can contain a text. Specializes VaerdiType.	enhed (Klassid) = NULL vaerdi (string)	Character String (ST)	The character string data type stands for text data, primarily intended for machine processing (e.g., sorting, querying, indexing, etc.) Used for names,	

Name	Description	Attributes	HL7 data type	Definition	Comments
				symbols, and formal expressions.	
Boolean	W3C format.		BooleanNonNull (BN) specializes BL	*BN* constrains the Boolean type so that the value may not be NULL. This type is created for use within the data types specification where it is not appropriate for a null value to be used	
datetime	W3C format.		Point in Time (TS)		
float			Real Number (REAL)	Fractional numbers. Typically used whenever quantities are measured, estimated, or computed from other real numbers. The typical representation is decimal, where the number of significant decimal digits is known as the precision.	“A decimal representation, a floating-point register and a scaled integer are all possible native representations of real numbers for different implementation technologies.”
int			Integer Number (INT)	Integer numbers (-1,0,1,2, 100, 3398129, etc.) are precise numbers that are results of counting and enumerating. Integer numbers are discrete, the set of integers is infinite but countable. No arbitrary limit is imposed on the range of integer	

Name	Description	Attributes	HL7 data type	Definition	Comments
				numbers. Two NULL flavors are defined for the positive and negative infinity.	
long			Real Number (REAL)		An implementation of a Real.
string			Character String (ST)	The character string data type stands for text data, primarily intended for machine processing (e.g., sorting, querying, indexing, etc.) Used for names, symbols, and formal expressions.	

8 Care plan –

8.1 Introduction

This chapter is concerned with the mapping of the HL7 Care Plan R-MIM into GEPJ. The purpose of the analysis is to identify functional abilities and to compare the HL7 Care Plan model and the corresponding part of GEPJ, in order to be able to relate these to actual needs expressed by EHR users

8.1.1 Care Plan in HL7

Care plan is described in Cares Structures topic within the Care Provision domain in HL7.

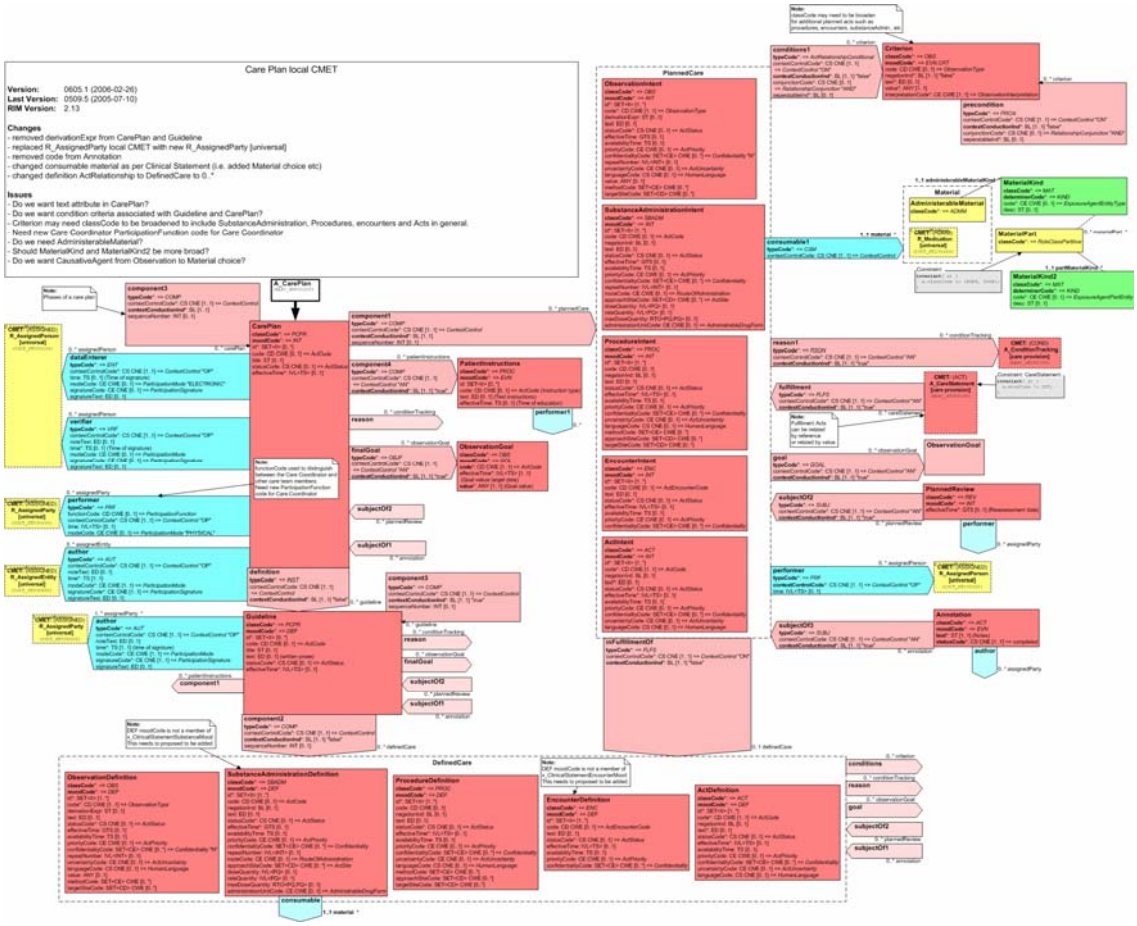


Figure 9 Care Plan in HL7

Core structures and concepts described in the R-MIM are

- CarePlan Act
- Component relationship
- PlannedCare ActChoice
- Guideline
- Goal
- Planned Review
- Patient instructions

- Annotation
- Criterion
- The different participations in CarePlan and PlannedCare

8.1.2 Care Plan in GEPJ

Care plan is not an individual concept in GEPJ. It is a certain number of states (“planlægningsfase” + “planlagt” + “udsat”) of an intervention.

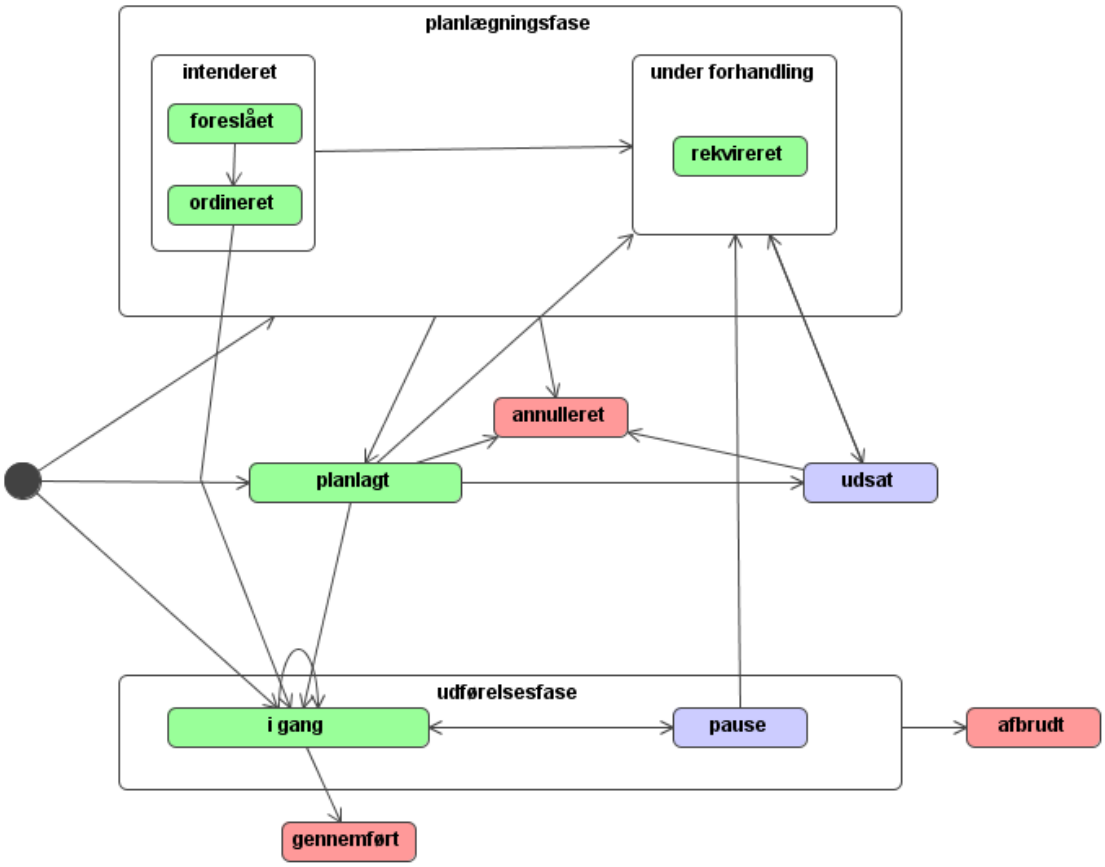


Figure 10 GEPJ Intervention states

This means that all functionality connected to a care plan is defined by the general concept of intervention.

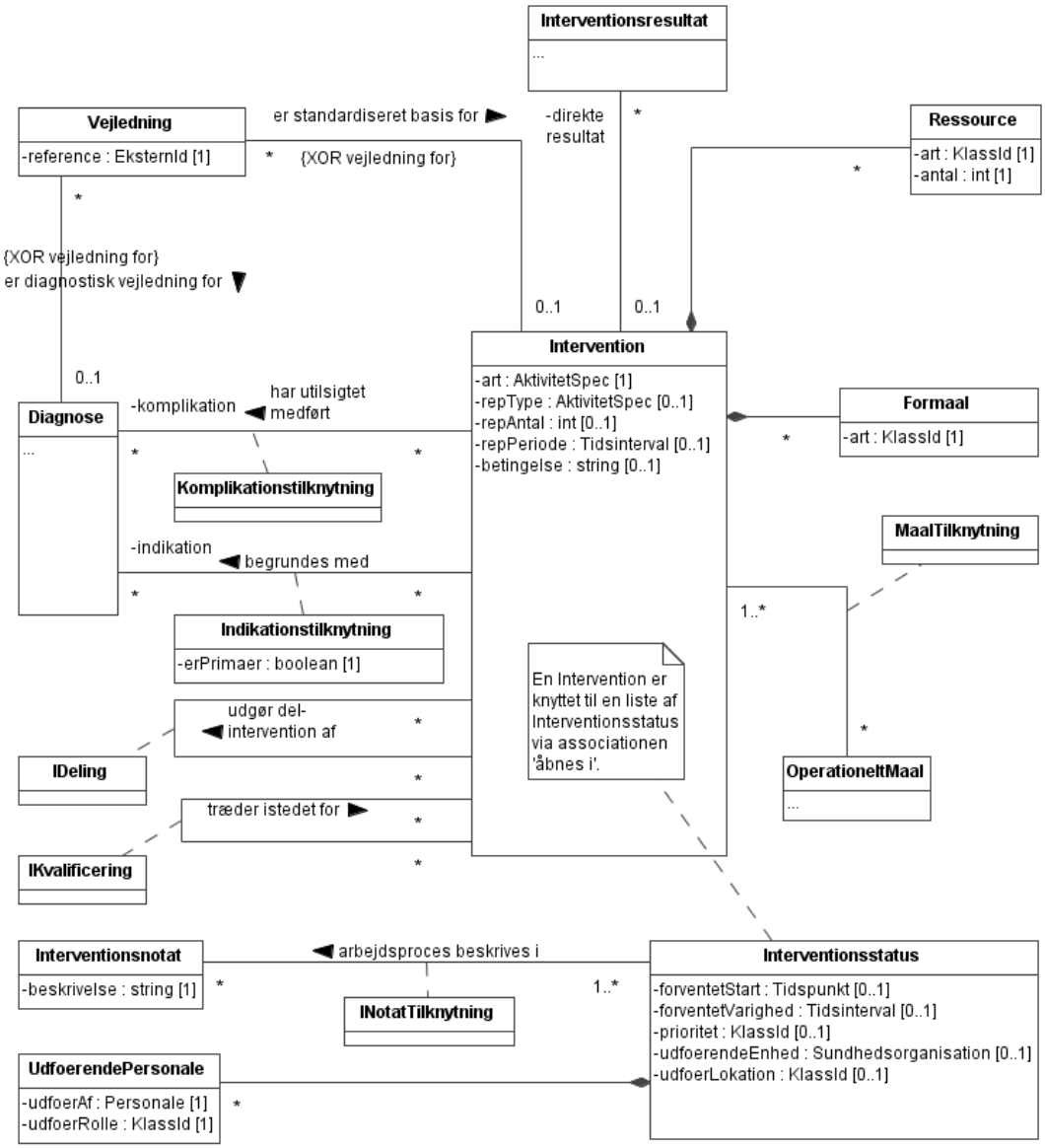


Figure 11 Intervention in GEPJ

8.2 CarePlan

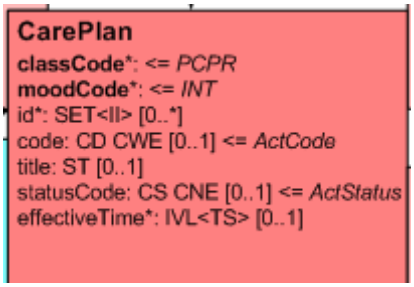


Figure 12 CarePlan core attributes

A CarePlan is mapped to an Intervention in GEPJ.

Attributes

HL7	GEPJ	Comments
Mood	Status	More submoods to INT than planning states of intervention.
Id	Id	Only one Id is possible in GEPJ. Multiple Ids are possible in HL7.
Code	Art	
Title		Not available in GEPJ
statusCode		Confers the general discussion on status differences between HL7 and GEPJ.
effectiveTime	ForventetStart, ForventetVarighed	More Complex Timing type in HL7.

8.3 Component Relationships

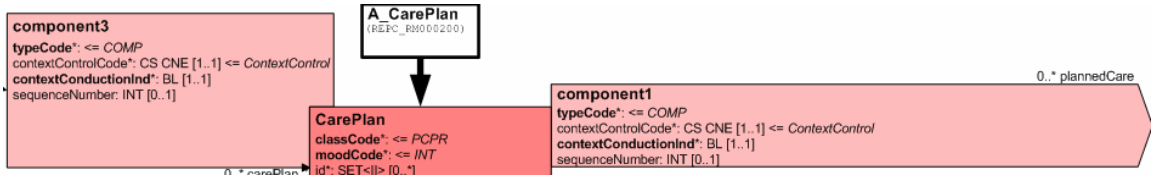


Figure 13 Component relationships in CarePlan structure

These relationships creates the hierarchical structure of CarePlan, where CarePlans may be build in multiple levels, eventually pointing to elementary actions, defined in PlannedCare acts.

The corresponding structure in GEPJ is 'Udgør delintervention af' association with the corresponding IDeling association class.

The major difference in modelling the hierarchical structure is the fact that GEPJ allows parent nodes in the hierarchy to be specialised classes as 'Patientkontakt' (encounter) and 'Medicinering' (SubstanceAdministration). In HL7 the only parent node is the 'CarePlan' node, which is an organiser of the plan.

Attributes

HL7	GEPJ	Comments
Type		Structural attribute in HL7
contextControlCode		Structural attribute in HL7
contextConductionIndicator		Structural attribute in HL7
SequenceNumber		Not available in GEPJ. This means that a predefined sorting of elements in a plan is not possible (except from sorting based on attributes for individual plan elements).

8.4 PlannedCare

This ChoiceBox is the elementary activities in a plan.

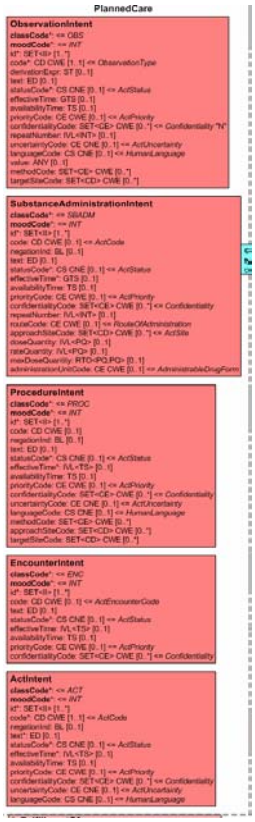


Figure 14 PlannedCare elementary activities

In GEPJ all these activities are modelled as interventions.

The main difference between HL7 and GEPJ is the number of different subtypes that are identified as possible elements in a plan.

Act types

HL7	GEPJ	Comments
ObservationIntent	Intervention	No subtype in GEPJ
SubstanceAdministrationInten	Medicinering	
ProcedureIntent	Intervention	No subtype in GEPJ
EncounterIntent	Patientkontakt	
ActIntent	Intervention	

8.5 Guideline

The guideline is defined as a standard care plan, and represents the corresponding master data for this plan.

Figure 15 Guideline section

GEPJ does not have any representation of structured guidelines or references from actual plans to structured guidelines. Instead GEPJ can refer to external guidelines.

8.6 Goals

Goals are available for a CarePlan as well as individual PlannedCare acts.

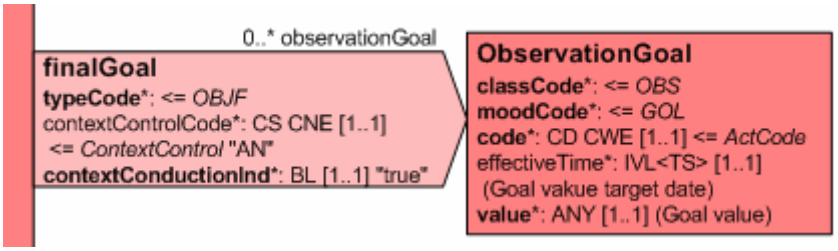


Figure 16 Goal related CarePlan. A similar construction exists on PlannedCare acts.

Goals are modelled in GEPJ in the ‘OperationeItMaal’ class.

8.7 Participations

A number of participations are connected to a CarePlan:

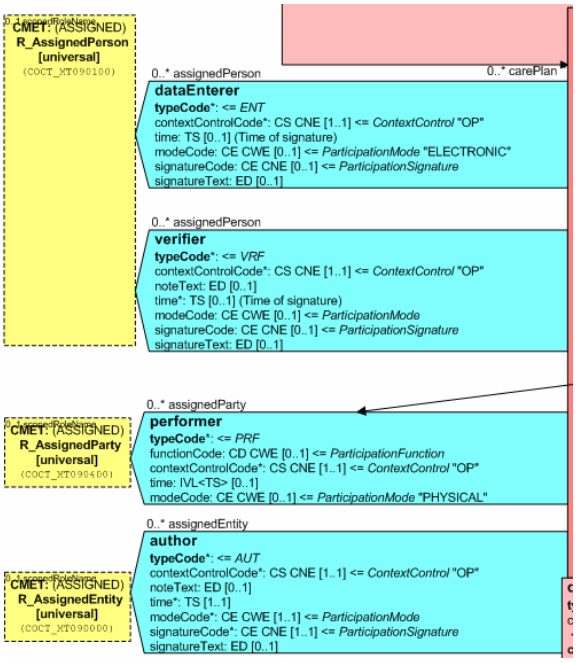


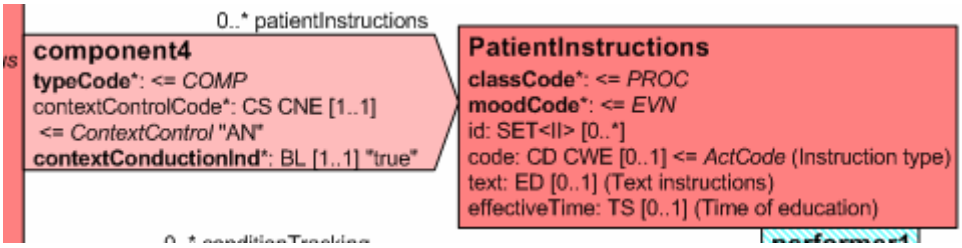
Figure 17 Participations on CarePlan

Participations in CarePlan

HL7	GEPJ	Comments
DataEnterer	Kontekst.dokumenteretAf, Kontekst.dokumenteret	GEPJ allows only one data enterer.
Verifier	Process data	Not available in GEPJ
Performer	Process data	Not available in GEPJ
Author	Kontekst.besluttetAf, Kontekst.besluttet	Only one author allowed in GEPJ. It is Not possible to have patient as author in GEPJ.

8.8 Patient Instructions

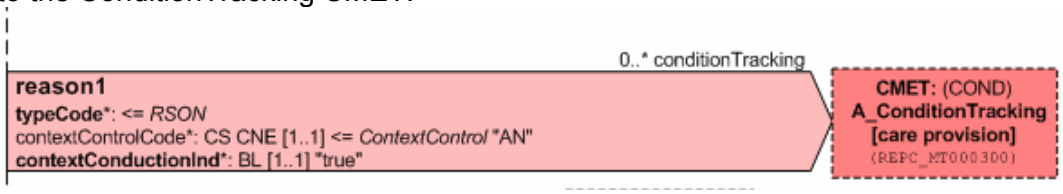
Patient Instructions holds the relevant instructions for the patient. The instructions that are derived from or associated with the guideline and the guidelines.



In GEPJ these are not modelled explicitly. They are expected to be ordinary interventions.

8.9 Reason (ConditionTracking)

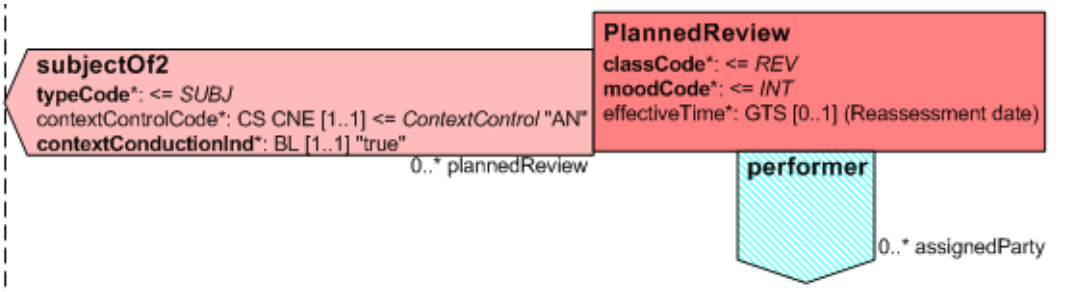
Via the Reason relationship the CarePlan and the PlannedCare choice box are linked to the ConditionTracking CMET.



This is modelled in GEPJ using the ‘indikation’ relation to ‘Diagnose’.

8.10 Planned Review

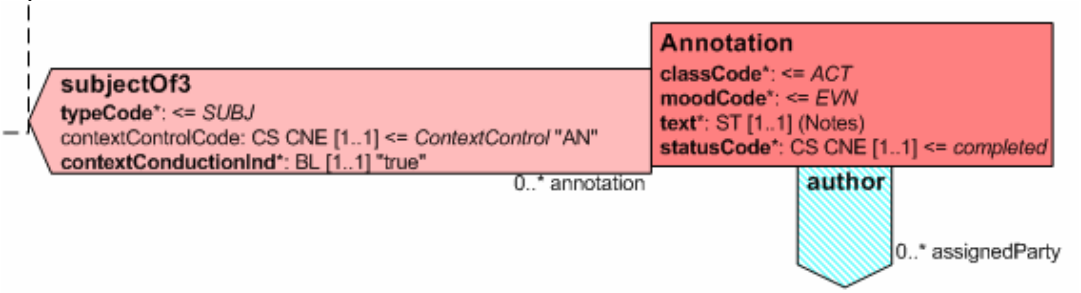
In the HL7 model you may define certain review points in your plan or in the individual acts.



GEPJ does not have a similar possibility.

8.11 Annotation

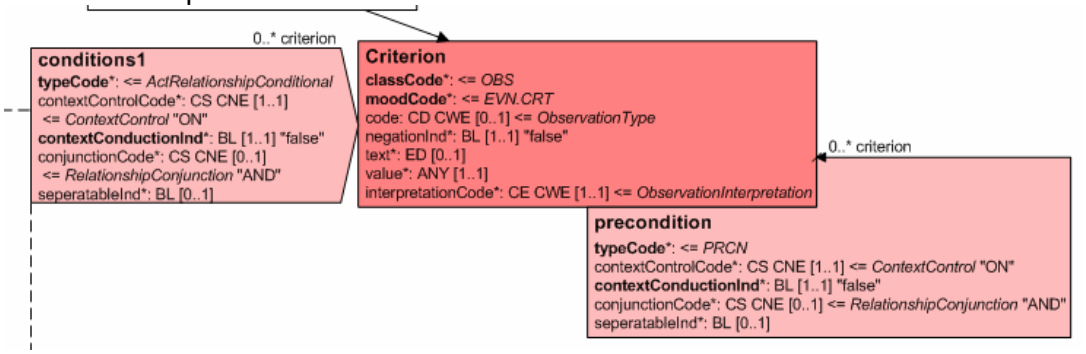
It is possible to add annotations to the CarePlan and to PlannedCare acts.



This corresponds to the ‘Interventionsnotat’ class in GEPJ.

8.12 Criterion

The criterion act relationship allows limits and criteria to be set up for particular acts. This makes it possible to define conditional execution of individual activities.



This corresponds to the attribute ‘betingelse’ on Intervention in GEPJ. This attribute is a text string, and as such a simpler construction than the HL7 construction.

8.13 Conclusion

The basic elements connected to a HL7 care plan are available within GEPJ. However the HL7 model has a number of elements not found in GEPJ.

Of 6 attributes, all clinically relevant attributes can be mapped.

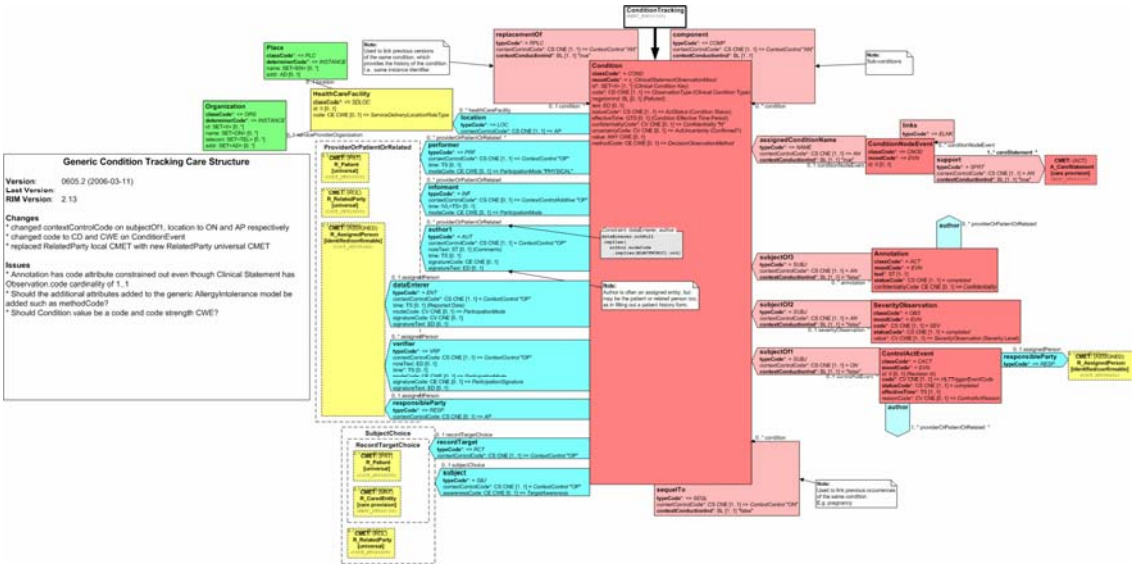
Of 5 classes, all can be mapped.

Of 4 relations, two can be mapped and two relate to process information.

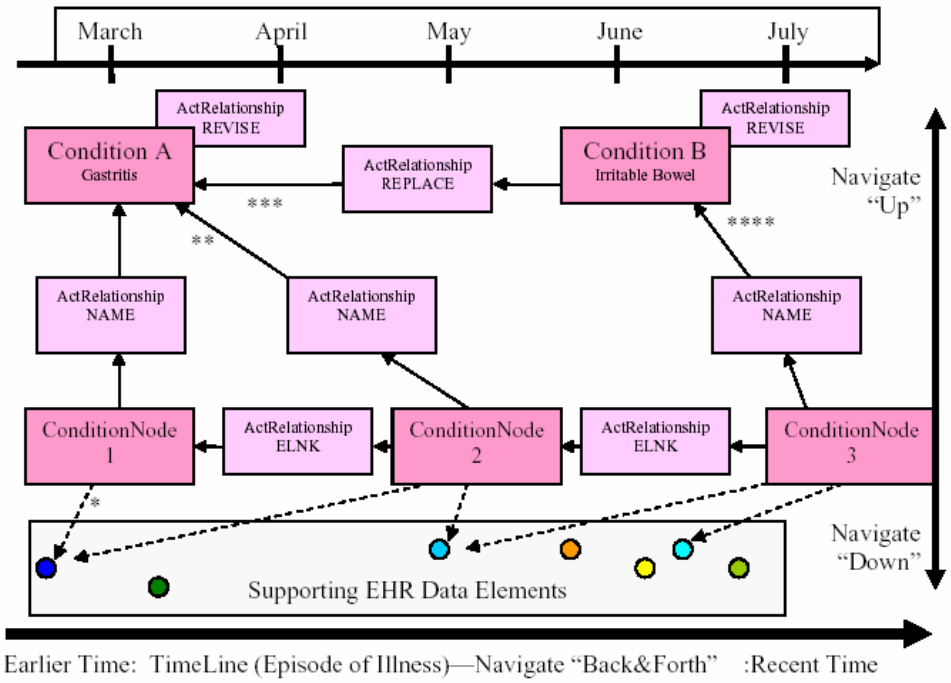
9 Condition tracking

9.1 Introduction

A selected part of the HL7 Condition Tracking logical model is mapped into GEPJ concepts in the following section. The method used is to take the concepts from the following figure (domains\pc\editable\images\REPC_RM000300.png in [1]) and identify the matching concept in GEPJ.



Another important figure is taken from [3]. It illustrates the dynamics of condition tracking.



As mentioned in the section on Fundamental Characteristics, the objectives of GEPJ and HL7 differ. In the following mapping, a comment about “process” or “non-clinical” means that the HL7 data item falls outside the GEPJ objective and this is not mapped.

9.2 Mapping Classes and Instances

HL7 Condition maps to GEPJ Diagnose and GEPJ DiagnoseStatus (see details later). A Condition documents the process of observing the condition of a patient. GEPJ only documents the result of this process.

The sequence of ConditionNodes is called an “Episode of Illness”. Each node represents the known information about one problem being diagnosed at a given point in time. The sequence of ConditionNodes maps to the GEPJ Forloeb. In GEPJ the relation Fokusering is related directly to the diagnosis. In HL7 the same relation is indirectly related to the condition through the ConditionNode.

The instances used differ significantly. GEPJ makes updates to Diagnose by creating new instances of DiagnoseStatus. In GEPJ you can create relations without modifying the related objects. In HL7 objects can be updated and creating relations will update the relation source.

In [3] it is stated, that the history of updates in HL7 can be described using ControlActs. This is not very well documented in the rest of the standard.

9.3 Mapping HL7 Attributes

HL7	**GEPJ**	**Comments**
moodCode	Process	Only the mood Event is meaningful in GEPJ and therefore the attribute is not mapped.
Id	Id, status	Some practical problems may be expected since instances do not map one-to-one.
Code	Diagnose.Art	The diagnosis code is stored in Code or Value. The set of possible values must be restricted. HL7 allows any value, including text in the value attribute. GEPJ only allows coded values. The example in [3] indicates that the code/value can be updated. This is not a good idea. The replacementOf relation should be used.
NegationInd		GEPJ cannot handle a negated diagnosis. Such a diagnosis is not desired in GEPJ and should not be mapped.

HL7	GEPJ	Comments
Text	DiagnostikNotat.beskrivelse	
Statuscode	DiagnoseStatus.art	Further restrictions on the codes are needed.
effectiveTime	DiagnoseStatus.opstaaet	If time is a range, GEPJ has no way to indicate the end time other than selecting state closed and use DiagnoseStatus.besluttet.
confidentiallityCode	Process data	Is not mapped to GEPJ. Concerns who can see information, which is important in information exchange, but is handled outside the data model in GEPJ.
uncertaintyCode	DiagnostikNotat.beskrivelse	GEPJ has no way of handling coded evaluation. Clinical evaluations are handled as DiagnostikNotat.
Value	Diagnose.Art	See Code
MethodCode	Proces data	Is not mapped into GEPJ. This concerns physical diagnostic equipment used in the process of defining a diagnosis.

9.4 Mapping HL7 Associations

HL7	GEPJ	Comments
Component	DAarsag (+ DDiff)	The semantic of sub-conditions in HL7 is vaguely defined. DAarsag and possibly also DDiff are visualized as a hierarchy in GEPJ and may therefore be considered components. Differentiating between DAarsag and DDiff may be based on Condition.uncertaintyCode. Mapping may change the clinical meaning.
replacementOf	DKvalisicering	
sequelTo		This is not mapped to GEPJ. The concept concerns previous instances of a similar diagnosis, e.g. if a bone has been broken before, the previous diagnosis for this is linked using this concept. Does not exist in GEPJ and should not be mapped.

HL7	GEPJ	Comments
assignedConditionName	ForloebsSymbolisering	Not an exact mapping.
ConditionNode.Support	Fokusering	GEPJ business rules do not allow this relation to be optional as it is in HL7.
subjectOf1		This is not mapped to GEPJ. Concerns historic relations. Unclear use, should be looked into.
subjectOf2	Axis in Tilstandsspec	This concerns severity of an observation.
subjectOf3	Diagnostiknotat	
Location	Process data	The closest concept in GEPJ is KontaktAnsvarsStatus.ansvarligEnhed.
Performer	Process data	GEPJ can only handle a cardinality of 1. If both a primary and a secondary performer are present, only the primary is mapped to GEPJ. HL7 allows a broader range of values, including the patient. Maybe Location could be used to get besluttetAf.tilknyttetEnhed.
Informant	Process data	This is not mapped to GEPJ. This concerns who has reported the information leading to the diagnosis.
Author	Process data	This is not mapped to GEPJ. This concerns who entered the information into the system.
dataEnterer	Dokumenteret, DokumenteretAf	GEPJ can only handle a cardinality of 1. GEPJ insists that the time must be system time. Signature information is NA. ModeCode is NA. Somehow the dataEnterer.time overlaps with act.availabilityTime. GEPJ Dokumenteret may be closest to act.availabilityTime.
Verifier	Process data	This is not mapped to GEPJ. It concerns the process of verifying that a diagnosis has been registered correctly by e.g. a secretary.
responsibleParty	Besluttet, BesluttetAf	

HL7	GEPJ	Comments
recordTarget		This is not mapped to GEPJ. It concerns another patient related to the diagnosis, e.g. a reference to another patient that the information could be relevant for.
Subject	Patient	GEPJ can not handle the awarenessCode.

9.5 Conclusion

On a conceptual level, the parts of HL7 Condition Tracking that are relevant to a GEPJ model can be identified in GEPJ with only a few exceptions.

Of 11 attributes, only one attribute related to clinical information cannot be mapped. This attribute is actually not wanted in GEPJ (the concept of negation is thought to “litter” the medical record).

Of 17 relations, all but three clinical relevant relations are mapped. The three relations that cannot be mapped are non vital concepts that have no counterparts in GEPJ. It could be investigated further, if they are relevant to introduce in GEPJ.

Apart from these minor mapping problems, there are some attribute range and relation cardinality issues. The attribute range problem could be amended by restricting the used codes and moods of HL7. The relation cardinality issues are e.g. that some relations are mandatory in GEPJ but not in HL7. Again, for a successful mapping to GEPJ, a constrained (national) HL7 usage is required as described in the chapter concerning fundamental characteristics.

10 References

[1] HL7 v.3 May 2006 Ballot.

[2] GEPJ v.2.2.

[3] HL7 v.3 May 2006 Ballot. Care Provision Domain Models: Explanation & Guidance. (domains/pc/ExplanationandGuidance.pdf).

[4] Grundstruktur for Elektronisk Patientjournal (GEPJ version 2.2 (20050812), Sundhedsstyrelsen 2005
http://www.sst.dk/applikationer/epj/gepj/022_20050812/index.html

[5] HL7® Version 3 Standard, © 2005 Health Level Seven®, Inc. All Rights Reserved.
Last Published: 03/26/2006 9:27 PM (May 2006 ballot)