HEARTFAID

D22 – Ontologies and knowledge representation

Submission date: 14/09/2007
Due date of document: 01/08/2007

HEARTFAID

A KNOWLEDGE BASED PLATFORM OF SERVICES FOR SUPPORTING MEDICAL-CLINICAL MANAGEMENT OF THE HEART FAILURE WITHIN THE ELDERLY POPULATION

Project summary	
Project acronym:	HEARTFAID
Project identifier:	IST – 2005 – 027107
Duration of the Project:	01/02/2006 – 31/01/2009
Project Co-ordinator:	UNICAL University of Calabria (Italy)
Thematic Priority:	Information Society Technology
Instrument:	Specific Targeted Research or Innovation Project

Consortium

- UNICAL - Università della Calabria (Italy)
- UNICZ - Università degli studi Magna Graecia di Catanzaro (Italy)
- UNIMIB - Università degli studi di Milano Bicocca (Italy)
- JUMC - Jagiellonian University Medical College (Poland)
- VMWS - Virtual Medical World Solutions Ltd (United Kingdom)
- FORTHNET S. A.- Hellenic Telecommunications and Telematic Applications Company S. A. (Greece)
- SYNAP - Synapsis s.r.l. (Italy)
- CNR - Consiglio Nazionale delle Ricerche (Italy)
- FORTH - Foundation for Research and Technology Hellas (Greece)
- RBI - Rudjer Boskovic Institute (Croatia)
- AUXOL - Istituto Auxologico Italiano (Italy)

D22 – Ontologies and knowledge representation

Document summary	
Document title:	D22 – Ontologies and knowledge representation
Document classification:	Derivable D22
Dissemination level:	PU
Submission date:	14 September 2007
Due date:	01 August 2007
Authors:	Antonio Candelieri, Domenico Conforti, Antonio Gualtieri, Vincenzo Lagani - UNICAL Domenico Costanzo - UNICZ Sergio Di Bona - SYNAP Ovidio Salvetti - CNR Franco Chiarugi, Ioannis Tsamardinos - FORTH Dragan Gamberger, Tomislav Šmuc, Mislav Malenica, Marin Prcela, Rajko Horvat, Tamara Švaglić, Alan Jović - RBI
Work package:	WP4 – Knowledge representation, discovery and management
Report version:	1.1

Short description
The document presents methodology for knowledge collection, formalization, and systematization and its concrete implementation on HEARTFAID tasks. The results are descriptive HF ontology, procedural knowledge base, HF medical plans, and ontological presentation of procedural knowledge. They are included in the electronic form on the CD that is part of this deliverable.

Change record		
Version number	Changes	Release date
0.0	first draft of the table of contents	24/05/2007
0.1	first draft of the document	20/06/2007
0.2	second draft of the document	09/08/2007
0.3	third draft of the document	14/08/2007
1.0	first version of the document	31/08/2007
1.1	final version	10/09/2007

Table of contents

1. Executive Summary

This deliverable presents the results of the task T4.4 "Ontologies and medical knowledge representation in the domain".

The document summarizes modern and potentially relevant approaches for medical knowledge presentation and describes requirements that a knowledge base should satisfy. Special attention has been devoted to different guideline modelling tools. The general conclusion is that more sophisticated methods are better for the expressiveness of knowledge representation but that the complexity of the reasoning process increases dramatically. Practically only relative simple representation forms without explicit time component can be effectively handled by available open source interpreters and reasoning systems. These are the reasons we have decided to use ontological form for the presentation of the descriptive knowledge and simple rule sets for the presentation of the procedural knowledge. The modern semantic web language (Web Ontology Language, OWL) has been selected as the most appropriate and has been systematically used in the development of the HF knowledge base. Protégé tool has been used for editing.

The information presented in the knowledge base has been obtained by human interpretation of guidelines for congestive and acute heart failure (http://www.escardio.org/knowledge/guidelines/), Heartfaid reports D5 and D9, as well as from other medical knowledge sources, including, but not limited to UMLS (Unified Medical Language System), Mayo clinic web site and Open Clinical web site.

The central part of the deliverable is presentation of the methods used to collect, systemize and formalize medical knowledge and presentation of the concrete results. The realized descriptive HF ontology includes around 200 classes, 2000 instances and more than 100 properties representing logical and functional connections among instances in different classes. The ontology is publicly available in the web form from the project web site (http://www.heartfaid.org/links.php). The procedural part of the knowledge base has been developed for HF diagnosis, HF severity assessment, treatment process, medication prescription and dosage, medication contraindications, prognosis estimation, and acute decompensation. The complete procedural part consists of more than 200 rules portioned in 10 subsets according to their target functionality. The problem of soft computing has been solved by a) various levels of reliability and/or probability of the rule outputs, and b) by complex deterministic computation of rule inputs during transformation from patient database to the ontological factual knowledge. The procedural knowledge base in the form of rules is in Appendix of this deliverable. Significant effort has been made to present the same procedural knowledge also in the ontological form. The resulting ontology can be found on the CD that is part of this deliverable.

The final part of the deliverable consists of two sections presenting challenges for further work and research. The first is integration of the developed knowledge base into the DSS system. Special attention is devoted to the problem of integration of the so called factual knowledge with real patient data from patient database into the ontological form. The significance of the approach is that all types of knowledge and information are in the unified ontological form prepared for direct interpretation and reasoning. The second part is presentation of medical plans which have been developed for the HF domain. Although strictly speaking not part of the developed knowledge base, they are a semi-formal systematization of the actionable medical knowledge. In its current form they have been used for the development of rules and for the verification of the resulting procedural knowledge base. It is the topic of the further research to test and demonstrate their usefulness as an intermediate step in knowledge base development and potentially as the third part of the knowledge base.

The deliverable also includes the CD with: descriptive HF ontology, implementation of procedural knowledge in the ontology form, complete set of developed medical plans in textual and graphical form, and Protégé tool for viewing ontologies.

2. Introduction

This document describes the results of the development of the HF platform knowledge base. Decision support subsystem (DSS) is the part of the platform responsible for its intelligent behaviour and the knowledge base is the representation of the medical knowledge necessary for the DSS operability. In order that this knowledge can be used by the DSS it must be presented in a formally sound way. The task of building the knowledge base consists of collecting the relevant medical knowledge, its systematization, and technical formalization.

User services are not supposed to directly access the knowledge available in the knowledge base. They can only ask for the assistance of the DSS, which can then decide to use the knowledge base for its decision making process. It means that during normal platform operation, the knowledge base, with exception of DSS, is isolated from other platform parts. In contrast to that, during the platform development the knowledge base is perhaps the most relevant integrative part between medical and technical partners. Building it presents the challenge of transferring all aspects of relevant medical knowledge into the platform. The success in this work significantly determines the overall performance of the system.

Medicine is the field characterized by the enormous amount of existing expert knowledge and at the same time there is a need for constant and reliable decision making. This is an ideal scenario for building and using automated knowledge based decision systems. Building an effective knowledge base is a challenge relevant not only for the HF platform, but for all artificial intelligence applications as well. It is also known as a hard problem with possibly many different solutions among which none can be selected as ideal or optimal for all situations. The knowledge representation is actually a very lively research field, especially in medical application. HF platform is a good example of a real medical environment for which automated intelligent decision making is necessary. In our work we first tried to test different knowledge representation options and then to select and use modern and most appropriate technology for solving concrete decision making problems.

3. Glossary of terms

ABox – a knowledge type in DL, holding a snapshot of the currently active problem

Arden Syntax – a language for encoding medical knowledge, based on MLM (rule-based)

Asbru – a task-specific and intention-based plan representation language to embody clinical guidelines and protocols as time-oriented skeletal plans

CDSS – Clinical Decision Support System

CUI – Concept Unique Identifier in UMLS

DBMS – Data Base Management System

description logics – a family of knowledge representation languages

descriptive knowledge – knowledge that describes the problem, entities involved in the problem, and relations between entities

DL – Description Logics

DSS – Decision Support System, a system that facilitates the decision making process

eD2R – a declarative language to describe mappings between relational database schemata and OWL/RDFS ontologies

ER – entity-relationship conceptual data model

factual knowledge – knowledge that describes the facts in a problem (e.g. patient data)

FMA – Foundational Model of Anatomy

frames – data structure used to divide knowledge into substructures

fuzzy logic – logic dealing with reasoning which is approximate rather than precisely defined

GALEN – Generalized Architecture for Languages, Encyclopedias, and Nomenclature in medicine

Glee – system for execution of guidelines encoded in the GLIF3 format

GLIF3 – Guideline Interchange Format, a computer-interpretable language for modelling and executing clinical practice guidelines

GMS – Guideline Modelling System

GMT – Guideline Modelling Tool

guideline modelling tools – methods and tools for the formalization of clinical practice guidelines

HL7 – Health Level Seven, standards for medical data storage and interchange

ICD – International Classification of Diseases

ICF – International Classification of Functioning, Disability and Health

inferencing – automatic process of deriving new knowledge form already defined one

Jess – a rule engine and scripting environment for Java platform

KAON – KArlsruhe ONtology, ontology infrastructure

KBDB-ETL – Knowledge Base - Database by Extraction, Transformation and Loading, database/ontology mapping tool

knowledge acquisition – a process of gathering knowledge from domain experts

knowledge base – a storage of formalized knowledge

LOINC – Logical Observation Identifiers, Names and Codes

MAPONTO – system for discovering semantic mappings between different data models

MedDRA – Medical Dictionary for Regulatory Activities

MeSH – Medical Subjects Headings

MLM – Medical Logic Module, storage for one rule in Arden Syntax

ONIONS – ONtological Integration Of Naive Sources

ontology – a data model that represents a set of concepts within a domain and the relationships between those concepts (a specification of conceptualization)

OTDs – Ontologies, Terminologies and Databases

OWL – Web Ontology Language, a language for defining and instantiating Web ontologies

patient record – a storage of patient related data

procedural knowledge – explicit knowledge necessary for realization of concrete tasks

production rules – IF condition THEN action items for representing knowledge

PRO*forma* – a logic- and task-based guideline representation formalism, grounded in well-defined logical model of decision making and plans

Protégé – ontology editor and knowledge acquisition system

Protégé -OWL – an extension of Protégé that supports OWL

R2O – extensible and semantically based database-to-ontology mapping language

RDMBS – relational database management system

reasoning – automatic process of deriving new knowledge from given concepts and facts

RIF – Rule Interchange Format, a language for encoding procedural rules

RIM – Reference Information Model, a model for expressing the data content needed in a specific clinical or administrative context, by HL7

RuleML – Rule markup language, XML representation of production rules

rules – production rules

semantic network – a directed graph consisting of vertices, which represent concepts, and edges, which represent semantic relations between the concepts

semantic web – an extension of WWW in which web content can be read and used by software agents

Snomed CT – Systematized Nomenclature of Medicine

soft computing – a collection of computational techniques for problems where conventional methods have not yielded low cost, analytic, and complete solutions

SPARQL – SPARQL Protocol and RDF query language

Swoop – a tool for creating, editing, and debugging OWL ontologies

SWRL – Semantic Web Rule Language, combining OWL and RuleML

Tallis – a Java implementation of PROforma-based authoring and execution tools

TBox – a knowledge type in DL, holding a terminology or taxonomy of the problem domain

UMLS – Unified Medical Language System

validation – checking whether the product design satisfies or fits the intended usage

verification – checking whether the system implements the specified functions

workflow – a reliably repeatable pattern of activity enabled by a systematic organization

4. Knowledge representation for medical applications

The fundamental goal of knowledge representation in artificial intelligence systems is to represent knowledge in the way that enables drawing useful conclusions. This section gives an overview of knowledge representation requirements and concepts in medical domains.

4.1. Medical knowledge base requirements

The medical knowledge base is an essential part of clinical decision support systems. The most of requirements set on the medical knowledge base are imposed by the requirements set on the clinical decision support system [1, 2]. An in-depth understanding of the clinical procedures is a crucial prerequisite for building the knowledge base. It should be followed by their precise and unambiguous description. The sources of errors in knowledge bases are misunderstanding of clinical procedures by the knowledge engineers and inadequate encoding of these procedures.

The central requirement is that the knowledge representation formalism used in the knowledge base must provide means for automatic reasoning. Selection of the appropriate knowledge representation formalism is a trade-off process because more expressive knowledge representation methods enable more effective encoding process but automatic reasoning is also more complex.

There are two different types of knowledge: descriptive knowledge and procedural knowledge [3]. Descriptive knowledge is conceptual knowledge about the domain. It is expressed in declarative sentences or indicative propositions. It describes the problem, entities involved in the problem, and relations between them. Procedural knowledge is actionable knowledge that can be directly applied to implement specific tasks in the domain. It describes procedures and actions. It is job specific and therefore less general than descriptive knowledge.

It is very important that the knowledge base provides means for handling both of these types of knowledge. The platform will apply procedural knowledge when selecting appropriate actions and descriptive knowledge when providing the context for them.

4.2. Knowledge acquisition

Knowledge acquisition can be defined as the process of eliciting, analyzing, transforming, classifying, organizing and integrating knowledge and representing it in a form which can be used in computer systems. Knowledge can be acquired from different sources: paper guidelines, the domain expert, or a group of domain experts [2, 3, 4, 5]. These sources differ in quality of knowledge they can provide but also in the complexity of the acquiring procedure.

With paper guidelines we already have a formalized knowledge in an explicit form. However, not all of the guideline's knowledge is represented explicitly. Guidelines are written by experts for experts, which means that they assume a vast amount of implicit knowledge. Guidelines often contain only recommendations based on scientific evidence, which is not enough to generate a clinical algorithm. For complete procedure construction, we must add information consisting of the expert's implicit and intuitive knowledge. Additionally, guidelines are not constructed in a way that reflects the flow of the real patient encounter. This makes the knowledge engineering process even more difficult.

Expert is a person with about ten years of full-time experience in his field of expertise. Expert has an extensive general knowledge of medicine and a deep, detailed knowledge of his relatively narrow areas of specialization. His knowledge is also hierarchical and densely interconnected. Experts are superior in perception of patterns. Working with an expert means we have huge amounts of knowledge at our disposal, but procedures for mapping it are much more complex than working with the paper guidelines.

Experts do make mistakes and don't know everything. Group of experts knows more than experts individually know and the group also tends to make smaller number of mistakes. Having many experts contributes to more robust system, because what one expert claims, others can check and correct. However, this additional feature demands even more complex knowledge acquisition procedures, because we have to be able to lead experts to a consensus and to resolve conflict situations [3, 6].

4.2.1. Knowledge types, the knowledge acquisition view

During the knowledge acquisition process we recognize two different types of knowledge: explicit knowledge and tacit knowledge. The difference lies in complexity of the elicitation process.

Explicit knowledge can be easily transmitted to others. It can be articulated into a formal language. It is expressed with words, mathematical and logical expressions. Examples are manuals and specifications.

Tacit knowledge is hard to encode with a formal language. It can be described as a personal knowledge embedded in an individual experience. It involves intangible factors as personal beliefs, perspective and value system. It consists of hunches, intuitions and subjective insights. It can be described through two dimensions: technical and cognitive. Technical dimension of tacit knowledge consists of know-how knowledge. This knowledge is derived from personal experience. Cognitive dimension is knowledge that is deep in us and we take it for granted. It consists of beliefs, perceptions, ideals, values, emotions and mental models.

Explicit knowledge is easily acquired by reading manuals, guidelines, specifications, through interviews with experts and so on. Tacit knowledge is much harder problem. Acquiring only one type of knowledge (usually only explicit) results with an incomplete and inadequate knowledge base.

4.2.2. Knowledge acquisition methods

Knowledge acquisition methods [3,4,7] can be divided into three groups: literature review methods, interaction with experts, and machine learning methods.

Literature review methods are very useful if the domain is well documented. Guidelines, manuals, specifications and scientific literature can contain large amounts of knowledge in an explicit form. This kind of knowledge acquisition must be used in conjunction with other methods because there is a great deal of implicit knowledge that is not written and therefore cannot be extracted in this way.

Interaction with experts can be done in different ways: straightforward interviews think aloud protocols, observational studies, group techniques, and computer based acquisition.

Straightforward interviews consist of asking a set of usually predefined questions. They require a minimum level of resources compared with other interaction methods. Although this kind of interviews can provide a large amount of qualitative knowledge, they have some disadvantages. Elicited knowledge frequently has a lack of quantitative data. Knowledge can be biased because of the presentation of questions or because of the selection of topics that are only of interest to researchers. Interviews most often lead to introspective opinions of collaborating experts, and the elicited knowledge may not correspond to what they actually do in real world situations.

To acquire relevant knowledge, we could observe experts in simulated or real world environments. We ask them to talk aloud about what they are doing and about what they are thinking, so we could get insights into their mental processes. This methodology is known as think aloud protocol.

In order to minimize researcher-induced biases, we can use observational studies as a knowledge acquisition method. This method minimizes knowledge engineer's involvement. We acquire information in a real world context and therefore as the result we have the situation specific knowledge. There are also disadvantages of this method: it is time consuming and it has problem with acquisition of quantitative information. Reasoning processes and knowledge structures must be inferred from collected information.

Working with a single expert has one major disadvantage that it is vulnerable to individual biases. Acquiring knowledge from multiple experts can reduce this problem, as well as enrich the knowledge base with multiple lines of reasoning and solve the problem of incomplete individual's knowledge. Derived knowledge is consensus based. There are a number of techniques for reaching consensus among the experts. This knowledge acquisition method has many advantages, but it is very difficult to conduct mainly because it is hard and expensive to organize multiple experts’ sessions.

Large scale decision support systems have an extensive and complex knowledge base and it is unreasonable to manage such an amount of knowledge manually. Specialized environments are developed to enable entering of new knowledge and maintenance of the knowledge base. Such computer-based knowledge acquisition systems usually rely on a domain knowledge base on which they construct procedural knowledge.

For very specific tasks and for managing historical data we can use machine learning methods to acquire knowledge from this data. The results are models that describe the relationships in the data and that can be used to construct knowledge structures and to describe reasoning logic.

4.3. Knowledge representation methods

Knowledge representation formalisms [1, 8, 9, 10, 11] for complex decision support systems are usually a combination of some basic knowledge representation formalisms. For example, ontologies are used to represent domain's descriptive knowledge and rules are used to represent its procedural knowledge. State-of-the-art medical knowledge representation formalisms will be described in the next section. In this one we give a brief description of their basic building blocks [12].

4.3.1. Representation formalisms

There are three groups of knowledge representation formalisms: ontologies, probabilistic reasoning formalisms, and rules.

With ontologies we formalize a shared understanding of a domain. We enter definitions of concepts and relationships among them. Our goal is to enable software applications and humans to share and reuse the knowledge consistently. Knowledge is represented with some formal language and it allows logical inference, which provides decision support and explanation facilities.

In order to facilitate probabilistic reasoning we can use Bayesian networks, influence diagrams, or even decision trees. These are probabilistic graphical models used to answer probabilistic queries about its variables. They consist of decisions (alternative actions), state variables, preferences and relations among state variables. These relations can be probabilistic, logical or qualitative.

The simplest representation formalisms are rules. Rules can be used for expressing single medical decisions. They are very useful for alerts and reminders. To provide reasoning under uncertainty, fuzzy rules are created. Main differences are in the interpretation of the quantitative data, formulation of recommendations, and unequal importance of clinical indicators.

4.3.2. Guideline modelling and representation

Knowledge representation formalisms are built out of primitives. These primitives must be expressive enough to capture the various aspects of a guideline. Examples of primitives are: rules, nodes, and frames. With these primitives we must be able to handle patient data elements and to enter decision, action and time elements. Structural arrangement of these primitives must be complex enough to allow nesting, decomposition, branching and sequencing. If we are working with sequences, the temporal logic is also a prerequisite.

The representation should be supported by some formal language. This language should be expressive enough to capture all necessary information and relations but also simple enough to facilitate automatic reasoning and decision support within some time limits.

Multiple authors can develop and modify the knowledge base. In this way the knowledge base changes over time. Additionally, knowledge must be shareable among institutions and at the same time local adaptations must be possible. Local modifications must be stored separately so that the universal part can be shared and reused at other places.

4.4. Verification and testing

Clinical decision support system should be rigorously evaluated before widespread dissemination into clinical practice [13]. When preliminary testing suggests that a CDSS improves clinical care or patient outcomes, confirmatory controlled trials are warranted. If there are errors in the knowledge base there will be also errors in the performance of the decision support system.

4.4.1. Error origins

Errors come in different phases of knowledge base creation process and because of different reasons [4, 6, 7]. They are usually products of various biases. Bias is a skewing from a standard or reference point that degrades the quality of elicited knowledge. We recognize two types of biases: cognitive biases (thinking-based) and motivational biases (behaviour or personal agenda based).

Anchoring, confusion, underestimation of uncertainty, and availability are cognitive biases. Anchoring is demonstrated when expert cannot move from first impression thinking. Confusion because of differing assumptions or definitions or because of memory problems and fatigue also degrades the quality of acquired knowledge. Experts sometimes think they know more then they really do; this overconfidence is called underestimation of uncertainty bias. Availability bias is demonstrated in cases of rare events, because experts cannot accurately account for rare events.

Misinterpretation, wishful thinking and impression management are motivational biases. Misinterpretation is a result of knowledge engineer's inability to correctly understand and then adequately translate expert's knowledge. Wishful thinking is demonstrated when experts' hopes influence their judgement. Impression management bias occurs when someone is responding according to politically correct interpretations from social pressure or from individuals.

4.4.2. Requirements

Knowledge base must be unambiguous, as well as logically and semantically valid [13]. Logical validation checks how the rules and objects work together in order to reach logical conclusions. Knowledge base must be logically consistent and logically complete. Consistency means that contradictory conclusions cannot be drawn out of the knowledge base. Completeness means that for all possible input combinations conclusion can be reached. Logical consistency and completeness are prerequisites for semantic validation of the knowledge base.

The knowledge base is semantically valid if it is semantically complete and semantically consistent. Semantical completeness means that decisions are based on all information considered to be relevant by experts. Semantical consistency means that it is not possible to reach semantically contradictory conclusions.

4.4.3. Methods

To create a high quality knowledge base, verification and validation methods must be used before, during and after the knowledge acquisition process.

We use bias minimization techniques to inhibit the occurrence of bias caused errors. Interview's questions should be reviewed a number of times in order to be sure that they address relevant information, that they are unambiguous, appropriately worded and well structured. It is a good practice to select experts who represent a diversity of opinion and who don't have any interest in the outcomes of the analysis. Experts tend to answer complex or highly general problems by sacrificing details that could be important for resolution, so these problems should be decomposed into finer, more manageable details. Ethnographic techniques that make experts to think aloud and observe their behaviour in the real world environment minimize motivational biases.

There are several ways for validating knowledge models. We can use knowledge models from standard documents in the domain assuming that if they are standards they are automatically valid. We can create models with one expert and review knowledge with other experts. We can create models through joint development and consensus of a team of recognized experts in the domain.

We can use a true/false test to validate knowledge models. We ask a panel of experts whether models are true or false and analyze results statistically. More experts, the higher is the confidence level.

It is very hard to test for semantic incompleteness, but some clues exist. If there exist variables, statements, conclusions, etc. that are defined but not used, than it is probably because expert thought this knowledge will be useful to him but he never completed that part of the knowledge base.

Verification and testing resources are usually very limited. We test in simulated environment with a number of existing patient records. Output is validated by experts. We need to validate important rules (rules that cover many inputs) and critical cases. It is very important to pay attention to knowledge areas that are most controversial among experts and to watch for experts that disagree the most with their colleagues.

4.5. Integration into a clinical decision support system

Performance of the CDSS depends on the characteristics of the constructed knowledge base. Quality of knowledge is crucial prerequisite in order for CDSS to be accepted among users. Knowledge base must enable automatic reasoning within some time constraints but also it must provide means for knowledge evolution. Important is also the possibility to give explanations for the results of the reasoning process.

4.5.1. Integration obstacles

There are several reasons that can cause inability to integrate knowledge base into a decision support system: knowledge from the knowledge base is not interpretable by automatic parser or inference engine; knowledge representation formalism is too complex and CDSS does not meet execution time requirements; knowledge base does not support local modifications that enable regular work in local institution's environment; knowledge base is dependant on electronic medical record system which is not implemented in particular institution or it is not compatible with legacy applications.

If the knowledge fetched from the knowledge base fails to fit naturally into the routine process of care, practitioners will refuse to use CDSS. Sometimes it is only matter of poor human interface design or reluctance or computer illiteracy of some health care workers. Knowledge evolves over time and it is crucial for knowledge base to be up-to-date. If knowledge quality is unsatisfying, practitioners won't accept CDSS's recommendations and this will create a negative climate for acceptance of CDSS among users.

4.5.2. Requirements

Knowledge should be organized in such a way that it automatically prompts when something can be done. Level of rejection of CDSS system is much higher if practitioners must initiate system for every recommendation. It is better if CDSS gives direct recommendations rather than just assessments.

The philosophical underpinnings of a medical knowledge base are important. If the sole purpose of a knowledge base is to support the function of a medical decision support system, the most important metric of knowledge base content is system behaviour. If, on the other hand, the purpose of the knowledge base is to serve as an academic repository of diagnostic information, then system performance, while clearly important, may be more secondary.

Computerized medical knowledge bases must be revised constantly, and can never be considered completely finished. The area of KB research known as "knowledge refinement" typically focuses on correction of KB errors after initial release of an expert system when it fails to perform as expected. Much of this work has been confined to non-medical domains [14]. However, this emphasis on error detection is present even in some work on medical KBs, such as GARVAN-ES1, an expert system used in the care of persons with diabetes [15].

As generally defined, knowledge refinement is a subcomponent of long-term maintenance. The emphasis in knowledge refinement is typically on locating and modifying individual, incorrect rules, generally in response to failures in the performance of the expert system. Long-term maintenance represents a considerable portion of the total life cycle of a medical knowledge base, involving three components: creation of new portions of a knowledge base; modification of existing portions of a knowledge base; and systematic checking to ensure consistency with externally available medical knowledge and better performance [16].

The type and structure of the knowledge base have important implications for the way in which long-term maintenance is performed. Systems that deal with very circumscribed domains may be able to rely on the knowledge of a single expert, and may require relatively limited changes over time. On the other hand, broad-scope medical knowledge bases generally use the published literature as their main knowledge source, and thus require a systematic, long-term process to track advances in the knowledge described in the literature. Such knowledge bases are often based not on rules, but on larger units of knowledge such as descriptions of entire diseases or pathophysiologic clusters that must be handled as a whole to ensure consistency.

The survival of a medical knowledge base over time is linked to its external anchors. If the knowledge sources are widely accessible, then the knowledge base is more likely to survive beyond its original creators and location.

4.6. Bibliography

[1] Enrico, C. (2003) *Clinical Decision Support Systems*. (2Rev ed.). Hodder Arnold.

[2] Peleg, M., Tu, S.W. (2006) *Decision Support, Knowledge Representation and Management in Medicine*. Haux & C. Kulikowski (Eds.): Schattauer.

[3] Shortliffe, E.H., Patel, V.L. (2006) *Human-Intensive Techniques*. R.E. Greenes (Ed.), Elsevier.

[4] Peleg, M., Gutnik, L.A., Snow, V., Patel, V.L. (2006) *Interpreting procedures from descriptive guidelines*. Journal of Biomedical Informatics, 39(2), 184-195.

[5] Tierney, W.M., Overhage, J.M., Takesue, B.Y., Harris, L.E., Murray, M.D., Vargo, D.L., McDonald, C.J. (1995) *Computerizing guidelines to improve care and patient outcomes: the example of heart failure*. J Am Med Inform Assoc., 2(5), 316-322.

[6] Shepherd, M.M., Wm. Benjamin Martz, J. (2004) *Group Consensus: Do We Know It When We See It?* HICSS '04: Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 1, Washington, DC, USA.

[7] Booker, J.M., McNamara, L.A. (2004) *Solving black box computation problems using expert knowledge theory and methods*. Reliability Engineering and System Safety, 85, 331-340.

[8] de Clercq, P.A., Hasman, A., Blom, J.A., Korsten, H. H. M. (2001) *Design and implementation of a framework to support the development of clinical guidelines*. International Journal of Medical Informatics, 64, 285-318.

[9] OpenClinical (2006) *Guideline modeling methods - Summaries*. URL: http://www.openclinical.org/gmmsummaries.html

[10] Peleg, M., Tu, J. (2002) *Comparing models of decision and action for guideline-based decision support: a case-study approach: Stanford University*.

[11] de Clercq, P.A., Blom, J.A., Korsten, H.H.M., Hasman, A. (2004) *Approaches for creating computer-interpretable guidelines that facilitate decision support*. Artificial Intelligence in Medicine, 31(1), 1-27.

[12] Wikipedia: *Knowledge representation*, URL: http://en.wikipedia.org/wiki/Knowledge_representation

[13] *Validating Underlying Knowledge*. URL: http://www.tfhrc.gov/advanc/vve/vve4a.htm

[14] Giuse, D.A., Giuse, N.B., Miller, R.A. (1995) *Evaluation of long-term maintenance of a large medical knowledge base*. JAMIA 1995(2), 297-306.

[15] Compton, P., Jansen, R. (1990) *Diabetes expert systems: planning for long term use*. Horm Metab Res Suppl, 24, 124-129.

[16] Jenders, R.A., Huang, H., Hripcsak, G., Clayton, P.D. (1998) *Evolution of a Knowledge Base for a Clinical Decision Support System Encoded in the Arden Syntax*. Proc AMIA Symp., 558-62.

5. Methods and tools: State of the art

Due to the broadness of the area, various approaches in knowledge representation have emerged. Every method is intended to solve different types of problems and has its own advantages and drawbacks. In this section we will analyze today's most relevant and the most referred methods and point out the ones that are well suited for the HEARTFAID platform.

5.1. Description logics

Description logics [1,2] are a family of knowledge representation languages which can be used to represent the terminological knowledge of an application domain in a structured and formally well-understood way. The name "description logic" refers to concept descriptions used to describe the domain and its logic-based semantics. It was designed as an extension of frames and semantic networks, which are not equipped with formal logic-based semantics.

The process of knowledge base realization by the use of DL involves precise characterization of the type of the knowledge to be specified by the system, as well as clearly defining the reasoning services the system needs to provide – the kind of questions system should be able to answer. Another aspect is providing an environment where the interaction with the KB will be used effectively. Most often, the interaction with the DL knowledge base is produced through the "Tell&Ask" interface, which enables knowledge base construction (entering the facts) and getting the information out of the KB.

Description Logic languages enable reasoning on the structured knowledge. Concept, as the basic DL building block, represents the generic entity that ensembles all the entities having similar behaviour and attributes. With respect to the first order predicate logic, concepts are unary predicates while roles are binary predicates.

Within a DL knowledge base, there is always a clear distinction between the:

- *Intensional* knowledge, which is a general knowledge about the knowledge domain. Intensional knowledge in the DL is comprised by the term TBox (where "T" stands for *terminology* or *taxonomy*).
- *Extensional* or *assertional* knowledge which is specific for the particular problem and for the individuals of the domain of discourse. In the DL it is comprised by the term ABox (where "A" stands for *assertion*).

The basic form of declaration in a TBox is concept definition which is performed by the definition of a new concept in terms of other, previously defined concepts. The basic task in constructing the terminology is classification, which puts the new concept expression in the proper place in the taxonomic hierarchy of concepts.

The basic reasoning tasks in the TBox are:

- *subsumption* – determine if one concept is subsumed by other;
- *equivalence* – deduce if one concept is equivalent to other;
- *satisfiability* – determine whether given concept is able to contain individuals without making a contradiction;
- *disjunction* – with given two concepts, determine whether any individual could be an instance of both of them at the same time;

The ABox contains extensional knowledge about the individuals of the domain of interest, for example stating that "Vito Gatuso" is a patient, or that "Vito Gatuso has symptom Headache". Assertions of the first type are called *concept assertions*, and of the second are called *role assertions*.

The basic reasoning tasks in the ABox are:

- *instance checking*, which is checking whether given instance belongs to a specified concept,
- *knowledge base consistency*, which is verifying that every concept in the base admits at least one individual,
- *realization*, which finds the most specific concept an individual object is instance of, and
- *retrieval*, which finds the individuals that are instances of the given concept.

It can be shown that all reasoning tasks can be reduced to a single one. The majority of today's available reasoners are designed to solve *satisfiability*.

Due to the trade-off between the expressiveness of the DL language and the complexity of reasoning with it, a careful selection of language constructs was needed. A specific DL language is named by assembling the letters that denote allowed DL constructs in that language:

- $\mathcal{AL}$ – attributive language. This is a base language which allows concept intersection, universal restrictions, atomic negation and limited existential quantifier;
- $\mathcal{FL}^-$ – a sub-language of $\mathcal{AL}$, disallowing atomic negation;
- $\mathcal{FL}_O$ – a sub language of $\mathcal{FL}^-$, disallowing limited existential quantifier;
- $\mathcal{C}$ – complex concept negation;
- $\mathcal{S}$ – abbreviation for $\mathcal{ALC}$;
- $\mathcal{H}$ – role hierarchies;
- $\mathcal{I}$ – inverse properties;
- $\mathcal{N}$ – cardinality restrictions;
- $\mathcal{Q}$ – qualified cardinality restrictions;

- $\mathcal{F}$ – functional properties;
- $\mathcal{E}$ – full existential quantification;
- $\mathcal{U}$ – concept union;
- $^{(\mathcal{D})}$ – datatype properties, data values and data types;
- $\mathcal{O}$ – nominals (enumerated classes of object value restrictions);
- $\mathcal{R}$ – limited complex role inclusion axioms; reflexivity and irreflexivity; role disjointness.

For example, OWL-DL provides the expressiveness of $\mathcal{SHOIN}^{(\mathcal{D})}$ description logics language.

The variety of DL languages led to the development of number of DL systems, all with the same basis but with different posed restrictions. The relevant description logics systems are IMACS, PROSE, LOOM, BACK and KRIS.

DL is already recognized as useful in the domains of medicine, software engineering, system configuration, web-information systems, digital libraries, and other. One focus of the research in medical environments has been on the construction and the maintenance of very large ontologies. The complexity of the domain (hundreds of thousands of concepts) led to the development of specialized systems, such as GALEN. The requirement for standardization led to the adoption of DL standard language KRSS in projects like SNOMED.

More recently there have been significant efforts based on the use of markup languages to capture the information content of Web structures. The relationship between DL and markup languages, such as XML, has been precisely characterized, thus identifying the DL language features for representing XML documents. The interest in standardization of knowledge representation mechanisms for enabling knowledge exchange led to the development of DAML-ONT, an ontology language for Web inspired by object-oriented and frame-based languages. Another language is OIL, a language with a similar goal of expressing ontologies, but with a closer connection to DL. Due to the similarities of above mentioned languages, the merge has occurred in the form of DAML+OIL language.

OWL (Web Ontology Language) [3] is a revision of the DAML+OIL web ontology language incorporating lessons learned from the design and application of DAML+OIL. OWL is intended to be used when the information contained in documents needs to be processed by applications, as opposed to situations where the content only needs to be presented to humans. OWL can be used to explicitly represent the meaning of terms in vocabularies and the relationships between those terms. OWL has more facilities for expressing meaning and semantics than XML, RDF, and RDF-S, and thus OWL goes beyond these languages in its ability to represent machine interpretable content on the Web.

SWRL (Semantic Web Rule Language) [4] is a language that is based on the combination of the OWL-DL and OWL-Lite sublanguages of the OWL with the Unary/Binary Datalog RuleML sublanguages of the Rule Markup Language. The proposed rules are in the form of an implication between an antecedent (body) and consequent (head). The intended meaning can be read as: whenever the conditions specified in the antecedent hold, then the conditions specified in the consequent must also hold.

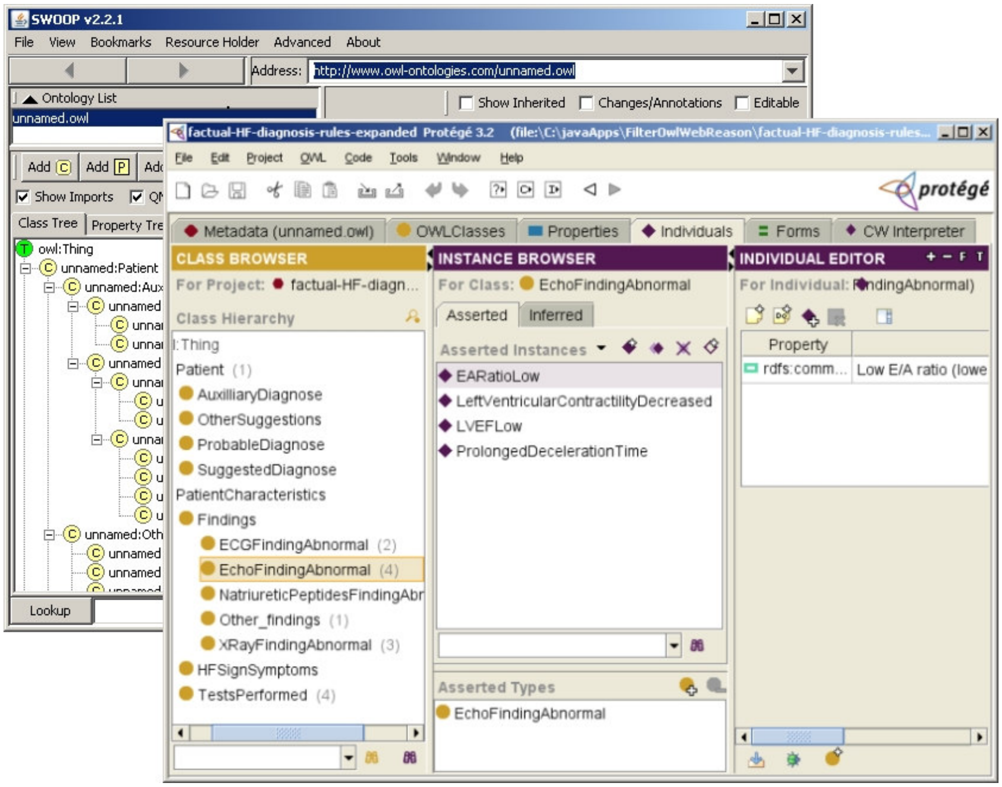


Figure 5-1. OWL editors: Protégé-OWL and SWOOP.

Direct practical results of the DL research are the tools for the construction of knowledge based applications, e.g. Protégé-OWL tool, and Swoop (Figure 5-1). The tools have rich graphical user interface for knowledge base construction and simple interface to the external reasoning tools. They also provide API-s for including the knowledge base into the applications. By the use of DL tools, knowledge bases are effectively constructed, maintained and simultaneously checked for their consistency.

5.2. Ontologies

Ontology [5] - the "science of being" - typically has different meanings in different contexts. Webster's Dictionary defines ontology as:

- A branch of metaphysics relating to the nature and relations of being

- A particular theory about the nature of being and the kinds of existence

Ontology can be viewed as a declarative model of a domain which defines and represents the concepts existing in that domain, their attributes and the relationships between them. It is typically represented as a knowledge base which then becomes available to applications that need to use and/or share the knowledge of a domain. Within health informatics, ontology is a formal description of a health-related domain.

The main purpose of ontology is to share common understanding of the structure of information between people and/or software agents, to reuse the domain knowledge and to make domain assumptions explicit.

In this sense we can speak of "ontology of cardiac valves" or "ontology of inflammation". Such ontologies are examples of the so-called "domain ontologies", whereas "foundational ontologies" represent domain-independent concepts like objects, events, processes.

The conceptualization of a domain of interest is usually performed through the definition of:

- Concepts
- Relationships
- Properties
- Objects
- Constraints

Ontology is a specification of a conceptualization carried out with:

- Formal notation
- Controlled vocabulary
- Documentation

The use of ontologies in medicine is mainly focussed on the representation and (re-)organization of medical terminologies.

In medicine, physicians have developed their own specialized languages and lexicons to help them store and communicate general medical knowledge and patient-related information efficiently. Such terminologies, optimized for human processing, are characterized by a significant amount of implicit knowledge. Medical information systems, on the other hand, need to be able to communicate complex and detailed medical concepts (possibly expressed in different languages) unambiguously. This difficult task can be achieved by constructing medical domain ontologies for representing medical terminology systems.

We have to consider that we can access a multitude of heterogeneous and autonomous data resources which differ in:

- Terminology

- Syntax
- Semantics

In healthcare systems we can identify the following "pro" and "contra" related to the use of ontologies. Positive aspects are:

- Ontologies can help build more powerful and more interoperable information systems in healthcare
- Ontologies can support the need of the healthcare process to transmit, re-use and share patient data
- Ontologies can provide semantic-based criteria to support statistical aggregations for different purposes
- Possibly the most significant benefit that ontologies may bring to healthcare systems is their ability to support the indispensable integration of knowledge and data

Negative aspect is that some practitioners remain sceptical about the impact ontologies may have on the design and maintenance of real-world healthcare information systems.

We strongly believe that the overall balance is in favour of using ontologies.

There are several kinds of ontologies like controlled vocabularies, simple taxonomies, classification hierarchies, semantic networkes, and logic based ontologies.

The common way of proceeding in the creation of ontologies is:

- Describe knowledge in terms of concepts and properties
- Define complex concepts in terms of other concepts and properties
- Use expressions to refer to complex concepts
- Create "is_a" relation between complex concepts
- Build up the model incrementally and descriptively
- Provide a reasoning mechanism

Ontologies can be built from scratch or taking advantage from existing Ontologies, Terminologies and Databases (OTDs).

The main existing OTDs in the medicine field that could be exploited for the heart failure domain are:

- Systematized Nomenclature of Medicine – Clinical Terms (Snomed CT)
- Unified Medical Language System (UMLS)
- Generalized Architecture for Languages, Encyclopedias and Nomenclatures in medicine (GALEN)
- Foundational Model of Anatomy (FMA)

- International Classification of Diseases (ICD)
- International Classification of Functioning, Disability and Health (ICF)
- Logical Observation Identifiers Names and Codes (LOINC)
- Medical Subjects Headings (MeSH)
- Medical Dictionary for Regulatory Activities (MedDRA)
- National Drug Code Directory
- ONIONS (*ONtological Integration Of Naive Sources)* Methodology
- Other OTDs

5.2.1. Snomed CT

Snomed CT has been developed by the College of American Pathologists & England and Wales National Health Service.

It is a generic healthcare terminology together with various relations between it's over 300,000 concepts. There are about a million descriptions of those concepts and about a million semantic links between them. The Snomed CT core content consists of:

- Concepts Table
- Descriptions Table
- Relationship Table
- History Table
- ICD Mapping

The main top classes consist of Clinical Finding, Procedure, Observable Entity, Body Structure, Organism, Substance, Pharmaceutical/Biologic Product, Specimen and Events.

Snomed CT classifies attributes according to the top classes. While some attributes are used across many top classes, there are many attributes characteristically used within a single top class. For example, Clinical Finding top class is associated with attributes like Severity, Onset, Course, Episodicity, Stage and so on. Similar, for Procedure, the attributes include Procedure Site, Procedure Device, Procedure Morphology, Access and so on.

Snomed CT is available under license for the countries within the European Union.

The Clue-5 tool is available for Snomed CT. Clue-5 is a Lookup engine for browsing SNOMED CT and for its integration with MS Windows-based clinical applications. The Clue-5 tool provides a reference and a browser server with an API for Snomed CT integration.

5.2.2. UMLS

UMLS has been developed by the National Library of Medicine [6].

It consists of Metathesaurus, Semantic Network and SPECIALIST Lexicon.

Metathesaurus is the vocabulary database of over a million terms dealing with the content of biomedical literature and Electronic Health Records. It consists of over 100 source vocabularies and tends to be univocal. If more than one meaning is assigned to a single vocabulary, then both the meanings of the term are represented within the Metathesaurus with the reference to specific source vocabularies. The source vocabularies integrated with the Metathesaurus includes ICD, Snomed, CPT codes, DSM, HUGO, MedDRA, NCI Thesaurus.

The Semantic Network consists of # Semantic Types that provide a consistent categorization of all concepts represented in the UMLS Metathesaurus as a set of Semantic Relations that exist between Semantic Types.

The SPECIALIST Lexicon provides the lexical information needed for the SPECIALIST Natural Language Processing (NLP) System.

UMLS is available under license for the users within the European Union.

UMLS resources are used in applications including information retrieval, natural language processing, creation of patient and research data and the development of enterprise-wide vocabulary services. NLM's applications include PubMed, the NLM Gateway, ClinicalTrials.gov and the Indexing Initiative. UMLS knowledge sources are distributed with flexible lexical tools and the MetamorphoSys install and customization program.

5.2.3. GALEN

GALEN has been developed by GALEN and the related European Union Project Participants.

The GALEN project developed a Common Reference Model for clinical terminology which can be applied to various medical domains. The GALEN project established the ontology and GALEN Representation and Integration Language (GRAIL) formalism and demonstrated the feasibility of the concepts; GALEN-IN-USE developed the Common Reference Model (CRM) for Medical Procedures - a key element for architectures for interworking between medical records, decision support, information retrieval and natural language processing systems in healthcare. OpenGALEN was established in 1999 as a not-for-profit organization to provide information on GALEN technologies and relevant software distributors and, in particular, to maintain and disseminate the CRM.

OpenGALEN is available for free use within the European Union within the terms of its license.

The available tools for GALEN are: Common Reference Model, GRAIL, Knowledge Management Environment (OpenKnoME) and GALEN Case Environment.

5.2.4. FMA

FMA has been developed by the Structural Informatics Group, University of Washington.

It is concerned with the representation of classes and relationships necessary for the symbolic representation of the structure of the human body in a form that is understandable to humans and is also navigable by machine-based systems. Specifically, the FMA is a domain ontology that represents a coherent body of explicit declarative knowledge about human anatomy. FMA has four interrelated components:

- Anatomy taxonomy: classifies anatomical entities according to the characteristics they share and by which they can be distinguished from one another.
- Anatomical Structural Abstraction: specifies the part-whole and spatial relationships that exist between the entities represented in the taxonomy
- Anatomical Transformation Abstraction: specifies the morphological transformation of the entities represented in the taxonomy during prenatal development and the postnatal life cycle
- Metaknowledge: specifies the principles, rules and definitions according to which classes and relationships in the other three components of FMA are represented.

FMA contains approximately 72,000 classes, over 115,000 terms and over 2.1 million relationship instances from 168 relationship types.

FMA is available for free use within the European Union within the terms of its license. A contract must be individually signed and a download access asked for.

An available tool for FMA is Foundational Model Explorer, which is an internet based FMA browser. FMA also allows StruQL queries which provide XML as output.

5.2.5. ICD

ICD has been developed by the World Health Organization.

It is designed to promote international comparability in the collection, processing, classification and presentation of diagnostics in health epidemiology, health management and mortality statistics. This includes analysis of the general health situation of population groups and monitoring of the incidence and prevalence of diseases and other health problems in relation to other variables such as the characteristics and circumstances of the individuals affected.

Top classes consist mainly of diseases classified according to the body system, though neoplasms, infectious diseases and injuries and poisonings have their own axes.

ICD is available for free use within the European Union within the terms of its license.

An available tool is ICD browser provided by the WHO. Many other browsers in different languages exist online.

5.2.6. ICF

ICF has been developed by the World Health Organization.

It is a classification of health and health related domains describing body functions and structures, activities and participation. The domains are classified from body, individual and societal perspectives. Since an individual's functioning and disability occurs in a context, ICF also includes a list of environmental factors.

The top classes of ICF are: Body Functions, Body Structures, Activities and Participation and Environmental Factors. Thus ICF provides terminology not just for functions, disability and Environmental factors, but also for the body structures, although they are not formalized and detailed like other ontologies e.g. FMA.

ICF is available for free use within the European Union within the terms of its license.

The same browser used for ICD (ICD browser, provided by the WHO) can be used for ICF. The classification of functioning and disability is useful to code patient status before and after therapy and also during the rehabilitation. ICF does provide a terminology which is useful for coding, however the classification is primitive and the relations between classes belonging to different axes does not exist. ICF's connection with ICD would improve the usage of both these terminologies.

5.2.7. LOINC

LOINC has been developed by the Regenstrief Institute and the LOINC committee.

It is a terminology primarily for laboratory results. It also covers certain kinds of clinical observations. It contains over 40,000 terms out of which over 30,000 deal with the laboratory domain. The laboratory portion of the LOINC database contains the usual categories of chemistry, hematology, serology, microbiology (including parasitology and virology), and toxicology; as well as categories for drugs, the cell counts and antibiotic susceptibility. The clinical portion of the LOINC database includes entries for vital signs, hemodynamics, intake/output, EKG, obstetric ultrasound, cardiac echo, urologic imaging, gastroendoscopic procedures, pulmonary ventilator management, selected survey instruments, and other clinical observations.

LOINC is available for free use within the European Union within the terms of its license.

A Windows-based mapping utility called the Regenstrief LOINC Mapping Assistant (RELMA) facilitates searches through the LOINC database and to assist efforts to map local codes to LOINC codes. Like the LOINC database, this program is also available for free use.

5.2.8. MeSH

MeSH has been developed by the National Library of Medicine.

It is a controlled vocabulary thesaurus consisting of sets of terms that are naming descriptors in a hierarchical structure which permits searching at various levels of specificity. The top-level classification includes: Anatomy, Organisms, Diseases, Chemicals and Drugs, Analytical, Diagnostic and Therapeutic Techniques and Equipment, Psychiatry and Psychology, Biological Sciences, Physical Sciences, and so on. MeSH is used on MEDLINE to index bibliographic citations and author abstracts from over 4,000 journals.

MeSH is available for free use within the European Union within the terms of its license.

MeSH Browser provides a searchable GUI for MeSH terms. PubMed uses MeSH as its terminology to search journal articles. HONSelect is a multilingual search tool which uses MeSH to link to various healthcare-related websites.

5.2.9. MedDRA

MedDRA has been developed by the International Conference on Harmonisation (ICH) and is currently owned by the International Federation of Pharmaceutical Manufacturers and Associations (IFPMA) acting as trustee for the ICH steering committee. It is maintained by MSSO - Maintenance and Support Services Organization.

It is a terminology for drug and medical device side-effects and malfunctions. It emphasizes on ease of use for data entry, retrieval, analysis and display when dealing with registering, documenting, and safety monitoring of medical products. The top-level classification of MedDRA consists mainly of disorders classified according to various body systems: Respiratory disorders, Cardiac disorders, Gastrointestinal disorders, Immune system disorders, Endocrine disorders, and so on.

An annual subscription (fee) is required for its use within the European Union.

MedDRA browser comes with the license agreement.

5.2.10. National Drug Code Directory

National Drug Code Directory has been developed by the Food and Drug Administration (FDA).

The Drug Listing Act of 1972 requires registered drug establishments to provide the FDA with a current list of all drugs manufactured, prepared, propagated, compounded, or processed by it for commercial distribution. Drug products are identified and reported using a unique, three-segment number, called the National Drug Code (NDC), which is the universal product identifier for human drugs. FDA inputs the full NDC number and the information submitted as part of the listing process into a database known as the Drug Registration and Listing System (DRLS). Several times a year, FDA extracts some of the information from the DRLS data base for publication in the NDC Directory.

NDC is available for free use within the European Union within the terms of its license.

No specific publicly available tools are provided.

5.2.11. ONIONS

ONIONS has been developed in the frame of an EU funded project.

It accounts for the problem of conceptual heterogeneity.

Aims of ONIONS include:

- Developing a well-tuned set of generic ontologies for supporting conceptual integration of relevant domain ontologies in medicine. Medical ontologies often lack semantic foundation, some axiomatization, or ontological depth.
- Integrating a set of relevant domain ontologies in a formally and conceptually satisfactory ontology library to support several tasks, including information access and retrieval, digital content integration, computerized guidelines generation, etc.
- Providing an explicit tracing of the ontology building procedure, in order to facilitate its maintenance (evaluation, extensions and/or updating, and intersubjective consensus).

ONIONS methodology exploits: a set of formalisms, a set of computational tools that implement and support the use of the formalisms, and a set of generic ontologies, taken from the literature in either formal or informal status and translated or adapted to the ONIONS formalisms.

The current main results of ONIONS are: the ON9.2 ontology library; the IMO (Integrated Medical Ontology) which represents the integration of five medical top-levels of relevant terminologies, and the relative mappings.

5.2.12. Other ontologies

Other relevant OTDs are:

- MedO – bio-medical ontology developed at the Institute of Formal Ontology and Medical Information Systems in Germany
- The ontology for the HL7 Reference Information Model (RIM)
- The terminology developed in the context of the IEEE 1073 family of standards

5.3. Production rules

A rule-based system consists of a set of IF-THEN rules, a set of facts, and some interpreter controlling the application of the rules, regarding the facts.

There are two broad kinds of rule system: forward chaining systems, and backward chaining systems. In a forward chaining system the reasoning starts with the initial facts, and keeps using the rules for drawing new conclusions (or for taking certain actions) given those facts. In a backward chaining system, the reasoning starts with a hypothesis (or a goal) that is being proved, and rule base is being searched for the rules that allow the hypothesis confirmation. Forward chaining systems are primarily data-driven, while backward chaining systems are goal driven.

If there is a particular goal (to test a hypothesis), then backward chaining will be much more efficient, as drawing conclusions from irrelevant facts is avoided. Forward chaining may be better if there are lots of things to prove (or if we just want to find out in general which new facts are true); when there is a small set of initial facts; and when there tends to be a lot of different rules, which allows drawing the same conclusion.

The rule syntax may vary but the expressiveness is about the same for all syntaxes. The execution speed, however, may vary a lot depending on the reasoner that is being used. Possible rule syntaxes are RuleML, RIF, Jess rule syntax, SWRL, etc.

5.4. Semantic networks

Semantic network is a directed graph consisting of vertices, which represent concepts, and edges, which represent semantic relations between the concepts. Semantic networks are a common type of a machine-readable dictionary.

The major idea is that:

- The meaning of a concept comes from its relationship with other concepts, and that,
- The information is stored by interconnecting nodes with labelled arcs.

5.5. Frames

A frame is an artificial intelligence data structure used to divide knowledge into substructures. Frames were proposed by Marvin Minsky in his 1974 article "A Framework for Representing Knowledge." [7].

Frames can also be regarded as an extension of Semantic nets. As tasks within a semantic network are becoming more complex, the representation needs to be more structured. A frame is a collection of attributes or slots and associated values that describes some real world entity. Frames provide way for encoding information to support reasoning. Each frame represents:

- a class (set), or
- an instance (an element of a class).

A frame system interpreter must be capable to implement following tasks in order to exploit the frame slot representation:

- Consistency checking -- when a slot value is added to the frame relying on the domain attribute and that the value is legal based on range and range constraints.
- Propagation of *definition* values along *isa* and *instance* links.
- Inheritance of default values along *isa* and *instance* links.
- Computation of value of slot as needed.
- Checking that only correct number of values is computed.

The most known tool for editing and authoring frames knowledge base is Protégé tool developed by Stanford Medical Informatics at the Stanford University School of Medicine. Protégé is open-source and freely available for use.

5.6. Guideline modelling methods and tools

General knowledge representation techniques are most often too simple for describing broad medical domains which have large amounts of complex knowledge. Knowledge bases built on general knowledge representation techniques are often user unfriendly and hard to maintain.

Guideline modelling tools [8,9,10,11,12] provide a framework for describing domain-specific knowledge by defining general domain-specific knowledge structures. Guideline modelling tools, when compared to general knowledge representation techniques, should ease the process of acquisition, verification, testing and maintenance.

The importance of using formalized medical clinical guidelines is widely recognized. However, the vast majority of them have failed to be used in practical implementations. The main reasons are:

- difficulties in acquisition and verification,
- difficulties in process of integration to medical institutions and
- inappropriate usage or usage denial by clinicians.

Among all of the guideline modelling tools, only the Arden Syntax [13] has been successfully implemented and integrated into the real medical environments. However, this does not deny the importance of the other tools; e.g. GLIF and Asbru still remain much more appropriate for modelling complex, specially temporal clinical workflows and plans than Arden Syntax.

One step towards the specific (medical) knowledge representation resulted in development of various approaches and techniques. Each approach is highly adopted to resolving one type of problem while other types remain hard or impossible to cover, e.g. Arden Syntax is appropriate for describing what actions should be taken in certain situation, but has major difficulties in describing medical treatment plans or sequence of actions. Also, guideline modelling tools are in general more adapted to solving particular small problems rather than covering the complete platform functionality. Solutions for large domain problems should be covered with combination (or hybrid) of a few guideline modelling tools.

The important issue concerning the usage of a specific guideline modelling tool in HEARTFAID platform is the availability of appropriate execution engine.

5.6.1. Arden Syntax

Arden syntax [13] is a rule-based system adopted in 1992. and is now part of HL7 standard. The main idea of the Arden syntax is to add as much as possible human-readable information to machine-readable rules. While the core of each rule remains machine-readable, additional information like plain-text purpose and explanation of the rule, author, date, specialist who approved rule usage, version and keywords are attached to the rule. This makes the process of rule creation, usage and maintenance much easier.

Each rule is stored in a single file and is called a Medical Logic Module (MLM). MLM's can be reused in many health institutions and platforms, although some adjustments have to be made to integrate them to each Health Information System. There are no commercially available components that enable integrating Arden into a new platform, which is the major drawback of the Arden syntax. The majority of the commercial applications are developed by vendors and are for use primarily within their own environment. Figure 5-2 gives an example of a single MLM that is suggesting a usage of ACEI inhibitor drugs in case when patient has a low LVEF.

```
MAINTENANCE:
Title:    Screen for ACEI prescription when patients LVEF is low
(triggered by LVEF storage)  ;;
Filename:   ACEI_when_low_LVEF ;;
Version:  1.00 ;;
Institution: Rudjer Boskovic Institute ;;
Author: Marin Prcela (marin.prcela@irb.hr) ;;
Specialist: ;;
Date:  2006-10-24 ;;
Validation: testing ;;

LIBRARY:
Purpose: Warn the health care provider to prescribe ACE inhibitors
if LVEF is lower than 45% ;;
Explanation: Whenever a LVEF value is stored it is checked whether
it is lower than 45%. If LVEF value is low and the patient is not
already using ACEI inhibitors patient health care provider is
alerted to consider prescribing ACEI.  ;;
Keywords:   LVEF; ACEI;;
Citations:   ;;
```

```
KNOWLEDGE:
Type:  data-driven;;
Data:
  /* evoke on storage of LVEF */
  storage_of_lvef := event {storage of LVEF} ;
  /* read the LVEF value that evoked the MLM */
  lvef := read last {LVEF value};
  /* read the if ACEI is already prescribed */
  acei_prescribed := read last {ACEI prescribed} ;
;;
/* MLM is executed when the storage_of_lvef is changed */
Evoke:   storage_of_lvef ;;
Logic:
     /* exit if the lvef value is invalid */
     if lvef is not number then
          conclude false;
     endif;
     /* exit if lvef is greater than 45 */
     if lvef >= 45 then
          conclude false;
     endif;
     /* exit if acei is already prescribed */
     if (acei_prescribed eq true) then
          conclude false;
     endif;
     /* send an alert */
     conclude true;
;;
Action:
       write "The patient's LVEF level  (" ||lvef|| "% on " ||time
of lvef|| ") is low and the patient is not taking ACEI. Therapy
using ACEI might improve LVEF value." ;
;;
Urgency: 50;;
END:
```

Figure 5-2. MLM example.

The drawback of Arden syntax is that it does not refer to any kind of domain description (ontology). Arden rules are built on the "high level", where variables are "high-level" concepts. Arden syntax does not provide any kind of bridge to the "low-level" parts of application; that problem is left open for the components that will integrate Arden to Health Information Systems. When a need for a specific term in MLM occurs (for instance reading the last measured value for patient's potassium level), it is placed in a curly braces and it is expected that this information will be fetched in some way. Arden does not pay attention to how it will be accomplished neither what is actually inside the curly braces. This problem is often referenced as a "curly braces" problem.

There are some tools available that enable building the MLM's database and simulate the execution of the rules. "Arden syntax checker" has the ability to check syntax of each rule, to interpret it and to simulate its execution. The underlying data fetching from the Health Information System is simulated by the user interaction.

Some of the HEARTFAID platform functionalities can be covered with Arden Syntax. In fact, rules are a perfect tool for watching over patient's signs and symptoms, for monitoring events and alerting medical experts etc. For instance, when patient's symptoms worsen (rule conditions fulfil), medical expert is warned instantly (rule action executes - message to doctors computer is sent).

Within the HEARTFAID platform, Arden syntax will require components for integration and for execution within the DSS of the platform.

5.6.2. The Guideline Interchange Format (GLIF)

GLIF [14] provides a framework for developing medical guidelines that are both easily understandable by humans (medical experts) and interpretable by machines. Each GLIF guideline is modelled in the form of a flowchart (directed graph). GLIF is suitable for describing logic sequence of actions.

GLIF chart (Figure 5-3) has four types of steps (nodes):

- action steps - used to model (1) tasks that should be performed in relation to patient (e.g. recommendation of a treatment), (2) programming-oriented actions (retrieving data from the EHR or sending message to the clinician) and (3) control-oriented actions (defining sub-guidelines or macros);
- decision steps - used to model decision points and to control flow of guideline executing;
- patient state steps - label which describes current patient state or which is used as guideline entry point;
- branch/synchronization steps - modelling parallel guideline execution or guideline concurrent choices.

GLIF provides three different levels of abstraction. At the conceptual level guidelines are represented as flowcharts that are easily understood or edited by humans. At this level medical knowledge is not interpretable by machines since the underlying platform implementation details are not formalized. This level provides graph structure and free-text description of each step in the graph. The computable level should provide the formal specification of each node with compliance to GLIF node structure. Implementation level should compile each defined variable from the computable level to its real value (e.g. data from medical institutions health record).

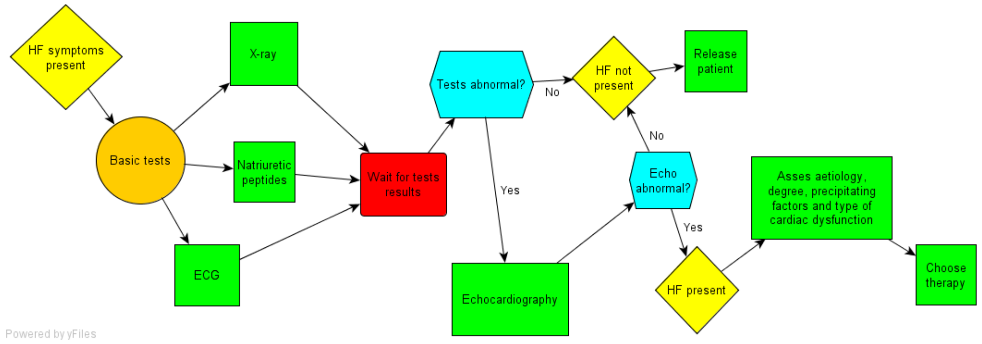


Figure 5-3. Example of encoded GLIF guideline for diagnosing Heart Failure disease.

One of the important GLIF features is its modularity - the ability to segregate one problem into more logically separated problems. This is accomplished by using sub-guidelines. GLIF also provides handling of exceptions and triggers. In this way the flow of the guideline execution can be abrupt and diverted to another branch that will handle specific medical situation (e.g. if blood pressure lowers to dangerous level, abrupt the current actions and proceed with handling emergency situation).

Each GLIF guideline is an ontology that is modelled following a specific GLIF structure. Therefore, any ontology editor may be used to model GLIF guideline (acquisition). Protégé ontology editor also provides a graphical user interface for editing flowcharts that are easily used for acquisition of knowledge at the conceptual level.

In order to provide easier integration into medical institutions, the patient data and medical concepts within the GLIF guideline can be defined in compliance with HL7 RIM standards.

Within the HEARTFAID platform GLIF may be used to represent the logical flow of actions, e.g. sequence of tests performed for diagnosing disease or prescribing therapy. GLIF will also require components for integration and for execution within the DSS of the platform, since there are no available execution solutions that are free (there is available commercial solution – Glee).

5.6.3. Asbru

Asbru [15] is a guideline modelling tool which focuses on representing medical plans. It is highly aware of the time dimension in the medical procedures and actions.

A plan in Asbru is a set of actions that are performed when certain preconditions hold. Each plan is decomposed into more sub-plans that are performed sequentially, concurrent (parallel execution) or cyclical. The plan that can’t be decomposed into sub-plans is considered an action. Figure 5-4 depicts the structure of plans for treating jaundice disease.

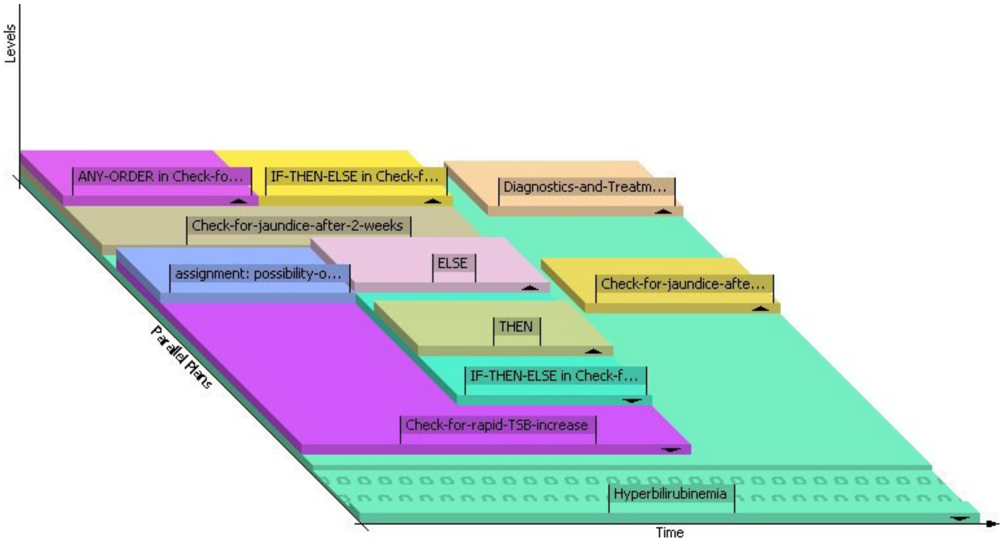

Levels
ANY-ORDER in Check-fo...
IF-THEN-ELSE in Check-f...
Diagnostics-and-Treatm...
Check-for-jaundice-after-2-weeks
assignment: possibility-o...
ELSE
Check-for-jaundice-afte...
THEN
IF-THEN-ELSE in Check-f...
Check-for-rapid-TSB-increase
Hyperbilirubinemia
Parallel Plans
Time

Figure 5-4. Preview of Asbru plan for treating jaundice presented in topological view.

Temporal aspects of the plan are specified by the following time stamps:

- earliest starting time,
- latest starting time,
- earliest finishing time,
- latest finishing time,
- minimal plan duration and
- maximal plan duration.

Not every time stamp has to be defined for each plan; e.g. plan might be defined with only the earliest starting time and maximal plan duration. Figure 5-5 depicts the usage of time stamps in defining plan duration.

There are three types of conditions related to the plans:

- filter defines which plans are to be considered at the certain point in time,
- setup defines the initiating point of the plan,
- suspend/abort/complete is used to suspend/abort/complete the plan execution.

Conditions are also defined by using time stamps. For example, plan might be initiated if patient has been diagnosed a disease four to six months ago (using minimal and maximal duration time stamps).

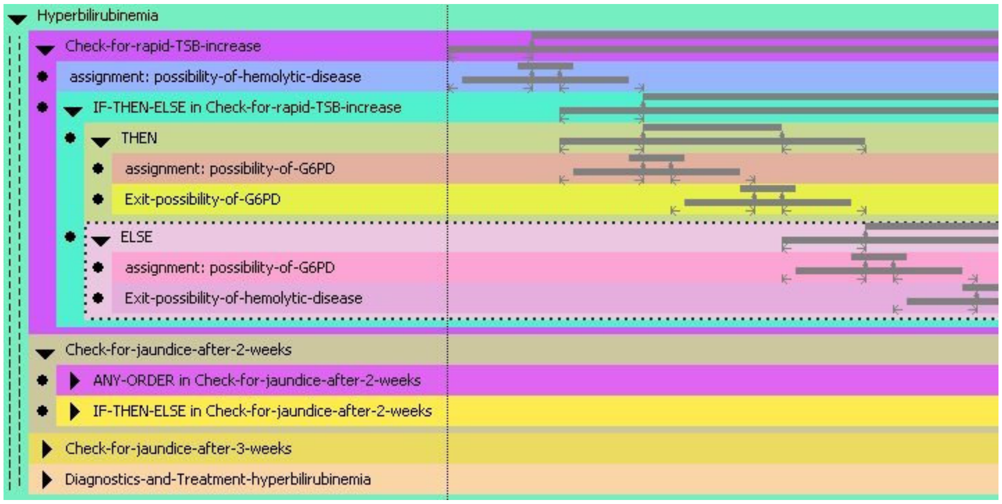

Hyperbilirubinemia
Check-for-rapid-TSB-increase
assignment: possibility-of-hemolytic-disease
IF-THEN-ELSE in Check-for-rapid-TSB-increase
THEN
assignment: possibility-of-G6PD
Exit-possibility-of-G6PD
ELSE
assignment: possibility-of-G6PD
Exit-possibility-of-hemolytic-disease
Check-for-jaundice-after-2-weeks
ANY-ORDER in Check-for-jaundice-after-2-weeks
IF-THEN-ELSE in Check-for-jaundice-after-2-weeks
Check-for-jaundice-after-3-weeks
Diagnostics-and-Treatment-hyperbilirubinemia

Figure 5-5. Preview of Asbru plan for treating jaundice in temporal view.

Each plan may have plan intentions defined. Intentions are high level goals of the plan and are describing what the plan is used for. If the intentions are defined, the clinicians may for some reason disregard the plan suggestion, as long as the defined intentions of the plan are being accomplished. For example, if plan suggests that patient should use beta-blockers with intention to lower blood pressure, clinician may disregard the suggestion if he plans to lower the blood pressure in some other way.

AsbruView is freely available tool for viewing and editing Asbru guidelines. It provides user friendly graphical interface available in two main modes, topological and temporal. Asbru Interpreter is the tool that simulates the execution of Asbru guidelines and may be very helpful in the process of guideline acquisition, verification, and testing.

Within the HEARTFAID platform, Asbru can be used in situations where actions are taken in a predefined order, e.g. to describe the procedure at the baseline evaluation or additional patient visits to the clinics (take the patient data, anamnesis, family history, laboratory assessment ...). However, there are no freely available execution engines that may be integrated into HEARTFAID platform.

5.6.4. PRO*forma*

PRO*forma* is a knowledge composition language that aims to assist patient care through active decision support and workflow management. Similar to the GLIF model, it represents also guidelines as a directed graph in which nodes represent instances from the PRO*forma* task ontology. There are four types of tasks:

- plans – an ordered sequence of tasks which are executed (or aborted) if temporal and logical constraints are satisfied;
- decisions – a set of possible outcome candidates with logical expressions that support each candidate;

- actions – requesting enactment by an external agent (clinical user or device), e.g. sending a message to clinician;
- enquiries – used to acquire information (obtained by user or by software agent).

Guidelines are represented using the R^2L language, and transformed to the L_{R2L} language prior to execution. PRO*forma* contains a number of tools for developing guidelines. A major focus point is on guideline safety by defining additional safety-related operators such as integrity and safety constraints.

Considering the execution engines, Arezzo is a commercial version of PRO*forma*, while Tallis is a version available for educational and research purposes (under license agreement).

5.6.5. Other Guideline Modelling Tools

Other relevant guideline modelling tools are [12]:

- CPG-RA,
- EON,
- Gaston,
- GEM,
- Glare,
- Guide,
- Helen,
- HGML,
- Prestige,
- PRODIGY,
- SAGE,
- Stepper.

5.7. Choosing knowledge representation method within HEARTFAID environment

Knowledge representation methods can be compared in many aspects. The most relevant are:

- **expressiveness**
  - types of knowledge
    - descriptive
    - procedural

        - plans
    - levels of representation (conceptual to low level)
- **readability**
    - level of human-readability
    - level of machine-readability
- **maintenance** (ability to modify knowledge on the fly)
- **available solutions**
    - tools for building knowledge base
    - components for "executing" knowledge (reasoning)
    - integration with already developed knowledge (re-usability)
    - tools quality & support
- **compliance to standards** (HL7, UMLS, …)
- **integration** to real medical institutions and/or apparatus
    - interface to low level problems
    - tools for integration to medical institutions/apparatus
- **price**
    - money consuming (open-source / gnu public license / research and educational license / commercial product)
    - time consuming (real time execution)
- **status** (completed, under development, research phase)

In the process of building the platform it is crucial to precisely define its requirements. Once that task is completed, it is possible to see what features of each knowledge representation method can be used to best fit platform's needs. One method may seem more powerful than the other in various aspects, but the platform requirements should give the main reasons for choosing one method and disregarding the other.

The main stumbling stones in many knowledge representation methods (the ones that could possibly meet the platform requirements) are the lack of available solutions (tools) and their price (money consuming). Most of the solutions for reasoning are commercial tools (Jess, Glee, Tallis ...), without an alternative that is free. On the other hand, a portion of the non-commercial solutions are still in the research phase or under development, which cuts them off of the HEARTFAID candidate list.

Given all solutions described above, we strongly believe that the best way of building a knowledge base is by means of ontology. Ontology is a very powerful tool to store descriptive knowledge. It has a high level of human and machine readability. There are plenty of available solutions for building and reasoning upon the knowledge stored in the ontology format. One of the most important features of ontology is its reusability, which enables easy integration with other available solutions. Ontology also provides structured domain description, which acts like a bridge between "low level" parts of the platform and "high level" domain description knowledge (medical concepts, medical terms, actions …). Existence of well defined ontology somewhat eases the segregation and integration of the technical tasks and the medical tasks.

OWL ontology language, as the most modern syntax for building ontologies, is a language that highly resembles the HEARTFAID platform requirements, due to its high flexibility and expressiveness. The acceptance of OWL in artificial intelligence community made OWL one of the today's hottest research topics. A number of tools for building, maintaining and integrating the OWL knowledge base are freely available for use, and are also easily adoptable for the integration within the platform. The SWRL expansion of OWL brings in the additional reasoning features into the OWL syntax, without loss of OWL's generality.

5.8. Bibliography

[1] Nardi, D., Brachman, R.J. (2002) *An Introduction to Description Logics*. Description Logic Handbook, edited by F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider, Cambridge University Press, 2002, 5-44.

[2] Baader, F., Nutt, W. (2002) *Basic Description Logics.* Description Logic Handbook, edited by F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, P.F. Patel-Schneider, Cambridge University Press, 2002, 47-100.

[3] McGuinness, D.L., van Harmelen, F. (2004) *OWL Web Ontology Language Overview*. URL: http://www.w3.org/TR/owl-features/

[4] Horrocks, I. et al (2004) *SWRL: A Semantic Web Rule Language Combining OWL and RuleML*. URL: http://www.w3.org/Submission/SWRL/

[5] Noy, N.F., Rubin, D.L., Musen, M.A. (2004) *Making Biomedical Ontologies and Ontology Repositories Work*. IEEE Intelligent Systems, 19(6), 78-81.

[6] McCray, A.T (1993) *Representing biomedical knowledge in the UMLS semantic network*. High performance medical libraries: Advances in information management for the virtual era, 45-55.

[7] Minsky, M. (1974) *A Framework for Representing Knowledge*. MIT-AI Laboratory Memo 306.

[8] de Clercq, A., Blom, J.A., Korsten, H.M., Hasman, A. (2004) *Approaches for creating computer-interpretable guidelines that facilitate decision support*. Artificial Intelligence in Medicine 31(1), 1-27.

[9] Peleg, M., Tu, S., Bury, J., et al. (2003) *Comparing computer-interpretable guideline models: A case-study approach*. JAMIA 2003, 10(1), 52-68.

[10] Tu, S.W., Campbell, J.R., Musen, M.A. (2003) *The Structure of Guideline Recommendations: A Synthesis.* AMIA Annual Symposium, 2003, 679–683.

[11] Tu, S.W., Musen, M.A. (2001) *Representation Formalisms and Computational Methods for Modeling Guideline-Based Patient Care*. Proc First Europ Workshop on Computer-based Support for Clinic Guidelines and Protocols, 2000:125-142.

[12] *Open Clinical: knowledge management for medical care*. URL: http://www.openclinical.org/

[13] Hripcsak, G., Clayton, P.D., Pryor, T.A., Haug, P., Wigertz, O.B., van der lei, J. (1993) *The arden syntax for medical logic modules.* Forteenth Annual Symposium on Computer Applications in Medical Care, 200-204.

[14] Peleg, M., Boxwala, A., Ogunyemi, O. (2000) *GLIF3: The Evolution of a Guideline Representation Format.* Proceedings of the American Medical Informatics Association (AMIA) Annual Symposium. (20 Suppl):645-649.

[15] Shahar, Y., Miksch, S., Johnson, P. (1998) *The Asgaard project: a task-specific framework for the application and critiquing of time-oriented clinical guidelines.* Artificial Intelligence in Medicine, 14(1-2), 29-51.

6. Heart failure ontology

The heart failure ontology presents the formalized description of concepts for the whole heart failure domain. It includes basic HF concepts, properties that characterize patients, all relevant diagnostic examinations and tests, and treatment procedures. The ontology also includes other cardiovascular system related concepts as well as concepts related to other organs when they are connected with HF. The information presented in the ontology has been obtained by human interpretation of guidelines for congestive and acute heart failure (http://www.escardio.org/knowledge/guidelines/), Heartfaid reports D5 and D9, as well as from other medical knowledge sources, including, but not limited to UMLS (Unified Medical Language System), Mayo clinic web site and Open Clinical web site.

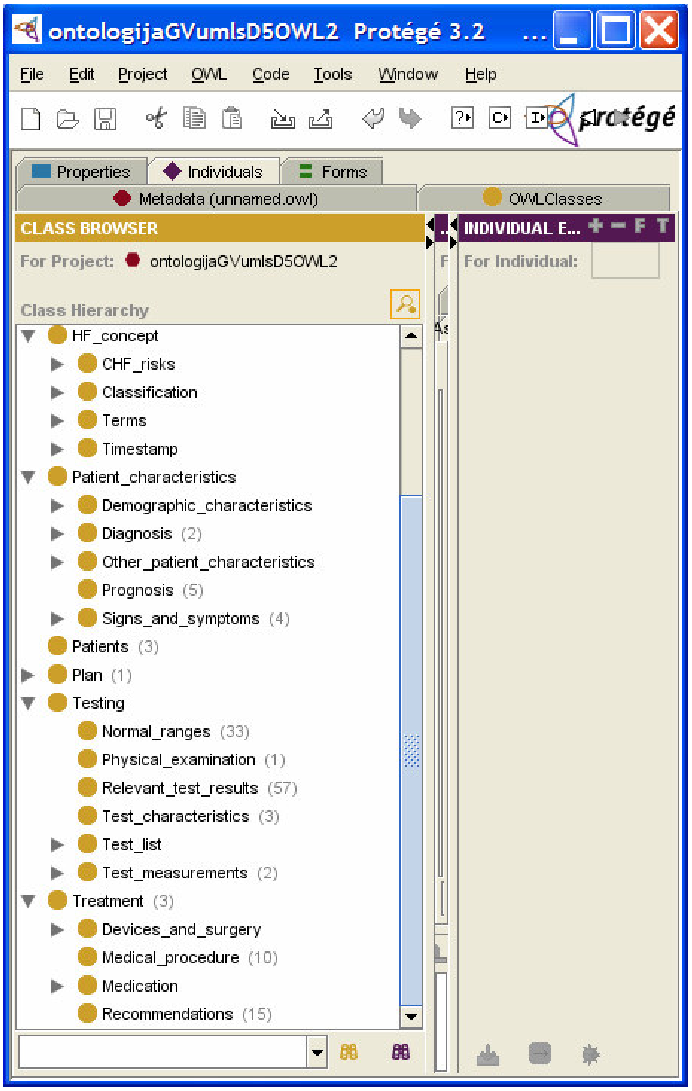
6.1. HF ontology description

In its current form the ontology presents the detailed taxonomic overview of the HF domain with around 200 classes describing HF related concepts. Examples are "Cardiac_hypertrophy", "Blood_pressure_signs" or "Heart_murmurs". These concepts are interconnected with super-class and sub-class properties into a hierarchical tree-like structure. At the basic level there are five relavant super-classes: "HF_concept", "Patient_characteristic", "Patients", "Testing", and "Treatment". Figure 6.1 presents the Protégé tool displaying these five super-classes with some of their most relevant sub-classes. Figure also depicts a super-class called "Plan" that is currently not relevant for the heart failure ontology.

Individuals or instances are members of the classes and typically present exhaustive list of concrete concepts relevant for the class. For example, the "Cardiac_hypertrophy" class has following six instances: "Cardiomegaly", "Combined ventricular hypertrophy", "Left_atrial_hypertrophy", "Left_ventricular-hypertrophy", "Right_atrial_hypertrophy", and "Right_ventricular_hypertrophy". The ontology includes more than 2000 individuals. When possible, classes are specified with their CUI number (Concept Unique Identifier according to UMLS) and with a list of synonyms. For example, for the class "Heart_diseases" its CUI is C0018799 and its synonyms are "Disorder_of_heart", "Cardiac_diseases", "Cardiopathy".

Finally, the ontology contains properties that connect individuals in different classes. These properties are relevant because they enable introduction of relations among concepts. For example, individual "Valvular_heart_disease" from the class "Heart_valve_diseases" is indicated by the individual "Dyspnea" from the class of "Signs_and_symptoms". Or that "Hyperkalemia" from the class "Potassium_disorder" may be caused by medications like "Potassium_sparing_diuretics" or "Spironolactone". The names of these properties are "Indicated" and "MayBeCausedByMedication". The HF ontology includes definitions of more than 100 properties.

The ontology presents descriptive domain knowledge. This knowledge is of two types. The first is defined by the generality relations among instances and classes, as well as by the generality relations among sub-classes and super-classes. In this way for each concept presented by some instance there is a series of *is-a* relation. For example, it means that "Cardiomegaly" *is-a* "Cardiac_hypertrophy" while "Cardiac_hypertrophy" *is-a* "Heart_disease". The second type of descriptive knowledge contained in the ontology comes from properties that define relations between classes, such as "Indicated" or "MayBeCausedByMedication", mentioned before.



- HF_concept
  - CHF_risks
  - Classification
  - Terms
  - Timestamp
- Patient_characteristics
  - Demographic_characteristics
  - Diagnosis (2)
  - Other_patient_characteristics
  - Prognosis (5)
  - Signs_and_symptoms (4)
- Patients (3)
- Plan (1)
- Testing
  - Normal_ranges (33)
  - Physical_examination (1)
  - Relevant_test_results (57)
  - Test_characteristics (3)
  - Test_list
  - Test_measurements (2)
- Treatment (3)
  - Devices_and_surgery
  - Medical_procedure (10)
  - Medication
  - Recommendations (15)

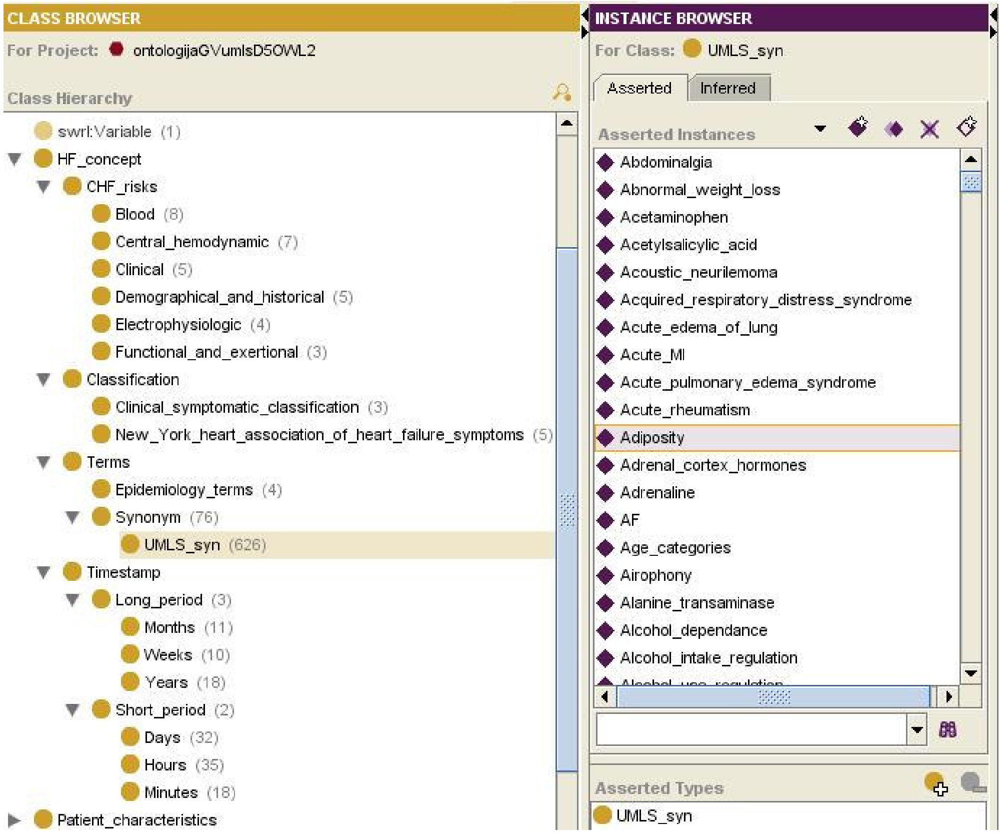
Figure 6-1 Protégé tool used to display a part of the HF ontology

The current version of the HF ontology is presented in the OWL (Web Ontology Language) form. This form is selected as identified in D15 as an appropriate ontological form for the realization of the decision support system using different query engines. Development and changes of the ontology are done by the publicly available visual interface.

Its textual form is in the XML format that is more than 0.5 MB long and is completely inappropriate for human inspection. There also exists a web form of the ontology generated as a series of HTML files from the original HF ontology. It can be freely accessed on http://lis.irb.hr/heartfaid/ontology/. This version is used for discussion and information exchange among project partners.

6.2. HF ontology structure

Five root classes in the heart failure ontology are "HF_concept", "Patient_characteristics", "Testing", "Treatment", and "Patients". Class "HF_concept" consists of hierarchy of classes which describe heart failure terminology, including the risks for congestive heart failure, medical synonyms, types of classification and also time frames used in treatment.



- swrl:Variable (1)
- HF_concept
  - CHF_risks
    - Blood (8)
    - Central_hemodynamic (7)
    - Clinical (5)
    - Demographical_and_historical (5)
    - Electrophysiologic (4)
    - Functional_and_exertional (3)
  - Classification
    - Clinical_symptomatic_classification (3)
    - New_York_heart_association_of_heart_failure_symptoms (5)
  - Terms
    - Epidemiology_terms (4)
    - Synonym (76)
      - UMLS_syn (626)
  - Timestamp
    - Long_period (3)
      - Months (11)
      - Weeks (10)
      - Years (18)
    - Short_period (2)
      - Days (32)
      - Hours (35)
      - Minutes (18)
- Patient_characteristics

Asserted Instances:

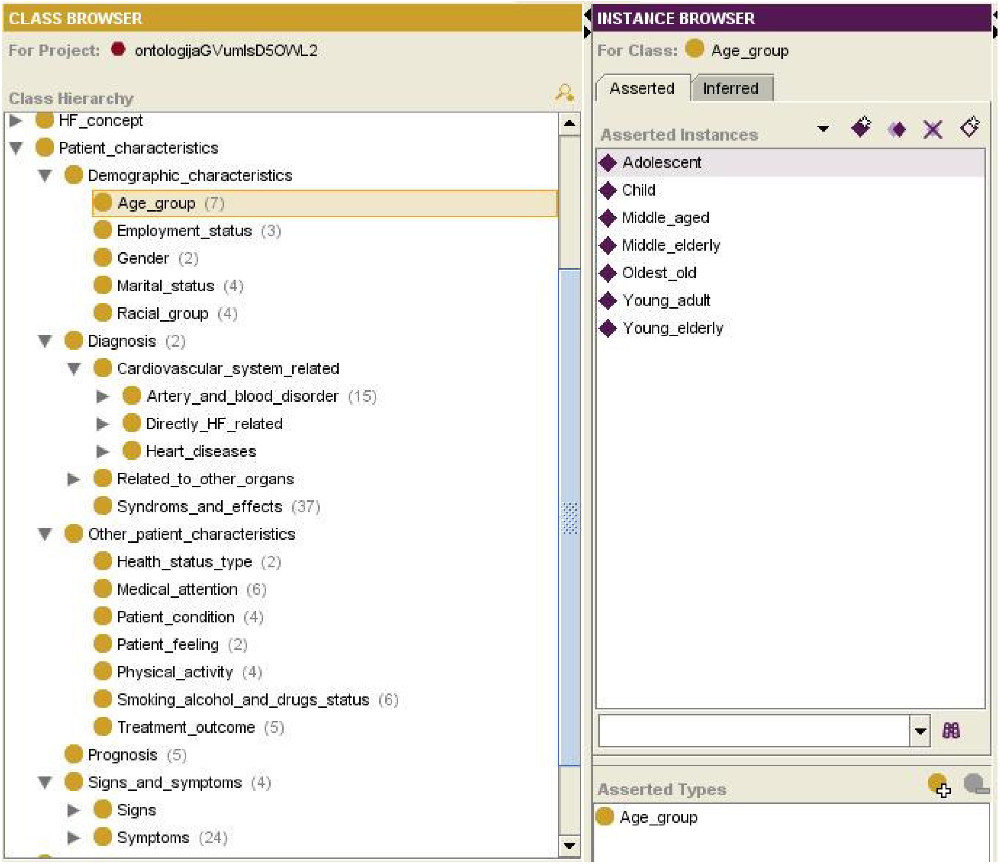
- Abdominalgia
- Abnormal_weight_loss
- Acetaminophen
- Acetylsalicylic_acid
- Acoustic_neurilemoma
- Acquired_respiratory_distress_syndrome
- Acute_edema_of_lung
- Acute_MI
- Acute_pulmonary_edema_syndrome
- Acute_rheumatism
- Adiposity
- Adrenal_cortex_hormones
- Adrenaline
- AF
- Age_categories
- Airophony
- Alanine_transaminase
- Alcohol_dependance
- Alcohol_intake_regulation
- Alcohol_use_regulation

Asserted Types: UMLS_syn

Figure 6-2. HF_concept class hierarchy

One can consider class "HF_concept" as a backbone of the ontology. It gives support and backup and also expands the ontology knowledge terms. The "HF_concept" part of the ontology is presented in Figure 6-2.

Class “Patient_characteristics” contains patient's demographical characteristics, possible diagnoses, possible signs and symptoms, prognosis and other characteristics. This hierarchy defines in fact the data in patient's heart failure medical record. It is interesting to note that both diagnosis and signs and symptoms have been placed in the class “Patient_characteristics”. These could have also been put on the first level of the hierarchy, since they are not patient specific. This placing is entirely arbitrary, but the goal was to not to have too many root classes. So, both diagnosis and signs and symptoms are thus considered as a patient’s intrinsic trait. The “Patient_characteristics” hierarchy is given in Figure 6-3.



CLASS BROWSER

For Project: ● ontologijaGVumlsD5OWL2

Class Hierarchy

- HF_concept
- Patient_characteristics
  - Demographic_characteristics
    - Age_group (7)
    - Employment_status (3)
    - Gender (2)
    - Marital_status (4)
    - Racial_group (4)
  - Diagnosis (2)
    - Cardiovascular_system_related
      - Artery_and_blood_disorder (15)
      - Directly_HF_related
      - Heart_diseases
    - Related_to_other_organs
    - Syndroms_and_effects (37)
  - Other_patient_characteristics
    - Health_status_type (2)
    - Medical_attention (6)
    - Patient_condition (4)
    - Patient_feeling (2)
    - Physical_activity (4)
    - Smoking_alcohol_and_drugs_status (6)
    - Treatment_outcome (5)
  - Prognosis (5)
  - Signs_and_symptoms (4)
    - Signs
    - Symptoms (24)

INSTANCE BROWSER

For Class: ● Age_group

Asserted | Inferred

Asserted Instances

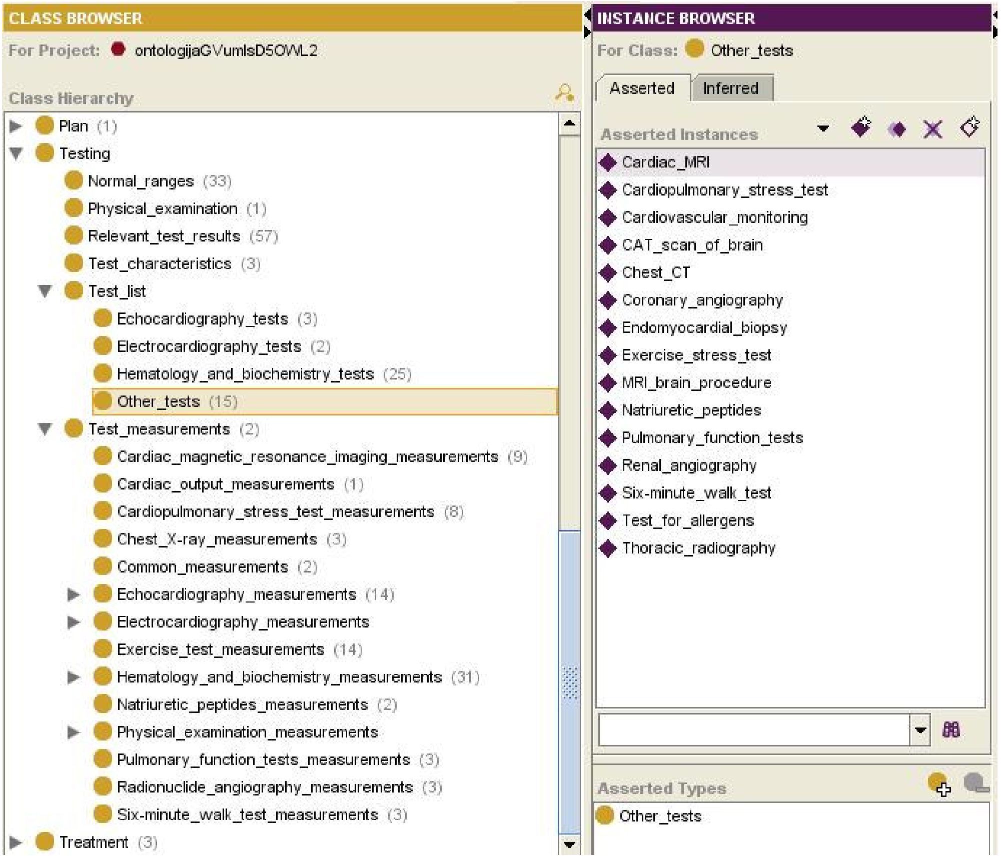
- Adolescent
- Child
- Middle_aged
- Middle_elderly
- Oldest_old
- Young_adult
- Young_elderly

Asserted Types

- Age_group

Figure 6-3. Patient_characteristics class hierarchy

Class “Testing” contains all the data regarding the tests performed in medical institutions. This includes a test list, usual measurements, measurements normal ranges and relevant results. Physical examination has also been placed within this class. Each test relevant to heart failure has properties that denote the measurements for that test and also which disorders it can detect. Some tests are invasive or used in combination with other tests and this information is also included. The specification of test measurements is as thorough as possible. The hierarchy of the class “Testing” is given in Figure 6-4.



CLASS BROWSER

For Project: ● ontologijaGVumlsD5OWL2

Class Hierarchy

- Plan (1)
- Testing
  - Normal_ranges (33)
  - Physical_examination (1)
  - Relevant_test_results (57)
  - Test_characteristics (3)
  - Test_list
    - Echocardiography_tests (3)
    - Electrocardiography_tests (2)
    - Hematology_and_biochemistry_tests (25)
    - Other_tests (15)
  - Test_measurements (2)
    - Cardiac_magnetic_resonance_imaging_measurements (9)
    - Cardiac_output_measurements (1)
    - Cardiopulmonary_stress_test_measurements (8)
    - Chest_X-ray_measurements (3)
    - Common_measurements (2)
    - Echocardiography_measurements (14)
    - Electrocardiography_measurements
    - Exercise_test_measurements (14)
    - Hematology_and_biochemistry_measurements (31)
    - Natriuretic_peptides_measurements (2)
    - Physical_examination_measurements
    - Pulmonary_function_tests_measurements (3)
    - Radionuclide_angiography_measurements (3)
    - Six-minute_walk_test_measurements (3)
- Treatment (3)

INSTANCE BROWSER

For Class: ● Other_tests

Asserted | Inferred

Asserted Instances

- Cardiac_MRI
- Cardiopulmonary_stress_test
- Cardiovascular_monitoring
- CAT_scan_of_brain
- Chest_CT
- Coronary_angiography
- Endomyocardial_biopsy
- Exercise_stress_test
- MRI_brain_procedure
- Natriuretic_peptides
- Pulmonary_function_tests
- Renal_angiography
- Six-minute_walk_test
- Test_for_allergens
- Thoracic_radiography

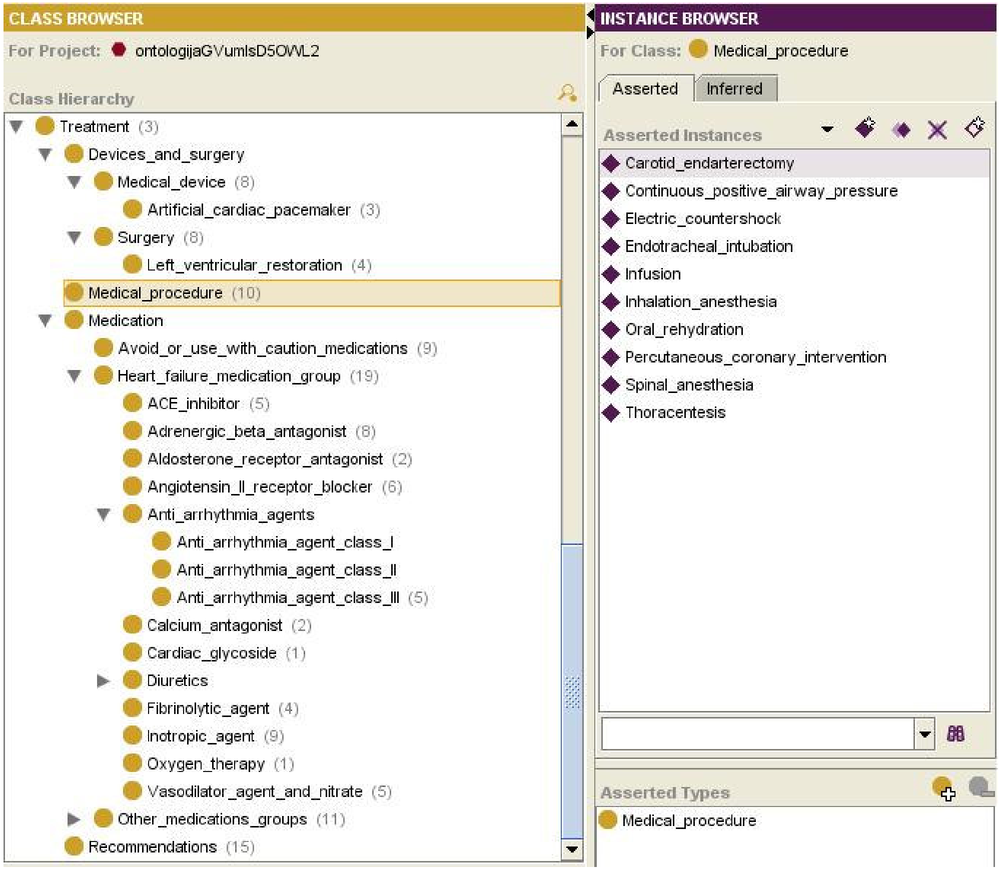
Asserted Types

- Other_tests

Figure 6-4. Testing class hierarchy

Class “Treatment” consists of medical procedures used in healing process, including medications, devices, surgery, procedures and recommendations regarding heart failure. Medications are presented by medication groups as well as by specific medications. Most of the medications relevant to heart failure symptoms and common comorbidities have been presented. Each medication has its dosage given and contraindications have been presented for medication groups. The hierarchy of the class “Treatment” is given in Figure 6-5.

Class “Patients” is an attempt to present the basic characteristics of patient's medical record. This class contains only individual patient's medical records, whose data is either received from the database or inferred using ontology reasoning.



CLASS BROWSER

For Project: ● ontologijaGVumlsD5OWL2

Class Hierarchy

- Treatment (3)
  - Devices_and_surgery
    - Medical_device (8)
      - Artificial_cardiac_pacemaker (3)
    - Surgery (8)
      - Left_ventricular_restoration (4)
  - Medical_procedure (10)
  - Medication
    - Avoid_or_use_with_caution_medications (9)
    - Heart_failure_medication_group (19)
      - ACE_inhibitor (5)
      - Adrenergic_beta_antagonist (8)
      - Aldosterone_receptor_antagonist (2)
      - Angiotensin_II_receptor_blocker (6)
      - Anti_arrhythmia_agents
        - Anti_arrhythmia_agent_class_I
        - Anti_arrhythmia_agent_class_II
        - Anti_arrhythmia_agent_class_III (5)
      - Calcium_antagonist (2)
      - Cardiac_glycoside (1)
      - Diuretics
      - Fibrinolytic_agent (4)
      - Inotropic_agent (9)
      - Oxygen_therapy (1)
      - Vasodilator_agent_and_nitrate (5)
    - Other_medications_groups (11)
  - Recommendations (15)

INSTANCE BROWSER

For Class: ● Medical_procedure

Asserted | Inferred

Asserted Instances

- Carotid_endarterectomy
- Continuous_positive_airway_pressure
- Electric_countershock
- Endotracheal_intubation
- Infusion
- Inhalation_anesthesia
- Oral_rehydration
- Percutaneous_coronary_intervention
- Spinal_anesthesia
- Thoracentesis

Asserted Types

- Medical_procedure

Figure 6-5. Treatment class hierarchy

6.3. Most significant classes

We will next present the examples of several important classes in heart failure ontology, including individuals and their relations by means of properties.

6.3.1. Class “UMLS_syn”

We first consider the class “UMLS_syn”, which is a subclass of class “Synonym”, subclass of “Terms” and subclass of “HF_concept” (see Figure 6-2). Class “UMLS_syn” contains many of the synonyms taken from Unified Medical Language System. Each individual of this class has its name and a property CUI, which specifies its number in UMLS. This class differs from a more general class “Synonym”, because it is necessary for an individual located in class “UMLS_syn” to be found in UMLS. The individuals in superclass “Synonym” need not be present in UMLS. If they are found in UMLS, than they also possess an unique CUI. Because of the implementation of OWL, there has to exist only one individual with a unique name in the whole ontology. Only one UMLS synonym for a term is chosen to be presented as an ontology concept. Most of other significant synonyms are placed in the class “UMLS_syn” and other synonyms for the same term are ignored. When one wishes to enter a new concept into the ontology regardless of its location, one also has to check if this concept is found in UMLS. If it is found, its UMLS name, usually the one given first in the UMLS listing of synonyms for a term, is added as a new concept (individual or class) in the ontology.

Other synonyms are added to the class “UMLS_syn” (with the same CUI, of course). If the term for given concept is not found in the UMLS, than the concept is added to the ontology without referencing to CUI. Sometimes, a term found in UMLS differs to a degree with the term for this concept searched by the author of the ontology. In such case, the term for a concept searched by the author is either discarded and the UMLS terms are used, or it is put in class “Synonym” as an additional term for the same concept and the UMLS term is used. For example, individual “Dehydration” of the class “Syndroms_and_effects” has UMLS synonyms in class “UMLS_syn”: “Exsiccosis” and “Dehydrated”. They all possess the same CUI: C0011175.

Another example is the individual “Acute_heart_failure” in the class “Heart failure”. It has synonyms: “Decompensated_heart_failure” and “AHF” in class “Synonym” and “Cardiac_failure_acute” in class “UMLS_syn”. Obviously, “Decompensated_heart_failure” and “AHF” do not exist in UMLS (at present), but do exist in guidelines for the acute heart failure and are thus added as synonyms for the acute heart failure (although decompensated heart failure is not strictly a synonym for the acute heart failure, it is its most common case).

6.3.2. Class “Diagnosis”

Class “Diagnosis” is a subclass of the class “Patient_characteristics”. It consists of three subclass: “Cardiovascular_system_related”, “Related_to_other_organs” and “Syndroms_and_effects”. “Cardiovascular_system_related” contains also three important subclasses as shown in Figure 6-3. These are “Artery_and_blood_disorder”, “Directly_HF_related” and “Heart_diseases”. Each of these classes are further divided into many subclasses and instances.

Class “Heart_diseases” contains the list of all heart-related disorders, excluding heart failure. For example, individuals “Left_atrial_hypertrophy”, “Myocardial_fibrosis”, “Cardiomegaly”, “Aortic_valve_insufficiency”, “Sick_sinus_syndrome”, “Left_bundle_branch_block”, and many others are members of class “Heart_diseases” and its subclasses.

Class “Directly_HF_related” contains very specific diagnoses related directly to heart failure, such as “Chronic_heart_failure”, “Acute_heart_failure”, “Left_ventricular_systolic_dysfunction”, “Diastolic_heart_failure” and others.

One of the most important classes in the ontology is “Artery_and_blood_disorder”, because heart failure and other heart problems are often in direct relation with the dynamics of the blood flow and also with its content. There are many blood disorders and not all are included in this ontology. Some of the examples are: “Acidosis”, “Hypovolemia”, “Sepsis”, “LDL_increased”, “Hyperbilirubinemia”, “Polycythemia”, “Thromboembolic_event”, “Hemorrhage”, “Hyperuricemia” and many others.

We had to add many other disorders that are not cardiovascular, because they are related to the functioning of the heart or are indicated as possible causes of the heart failure symptoms. These are included in the class "Related_to_other_organs". Some of the examples are: "Skeletal_muscle_problems", "Cerebral_hemorrhage", "Drug_abuse", "Multiple_sclerosis", "Hepatomegaly", "Cystic_fibrosis", "Pneumonia", "Pulmonary_edema", etc.

Some of the patient status that can not be exactly considered as a diagnosis and also some of the known syndroms are given in class "Syndroms_and_effects". The examples are: "Lack_of_adequate_sleep", "Meningism", "Overeating", "Reduced_sudden_death".

Class "Diagnosis" thus contains many different possible aspects of the heart failure disorder and even a wider scheme of diseases. Most of the significant diseases which are in any way considered in relation with heart problems are classified in this class.

6.3.3. Class "Medication"

Next, we consider class "Medication". This is a subclass of the root class "Treatment" and it contains almost every relevant heart failure related medication and medication group. It also contains some of the other medications used in treatment of heart related problems, such as medications for atrial fibrillation or high blood pressure.

This class is divided into three classes: "Avoid_or_use_with_caution_medications", "Heart_failure_medication_group" and "Other_medications_groups".

First class contains specific medications that should not be prescribed to the patient if the patient has heart failure, as recommended by chronic heart failure guidelines. The examples are: "Corticosteroids", "Diltiazem", "Verapamil", "Lithium".

Second class contains all of the relevant medication groups represented by both individuals and classes. Classes for the specific medication group contain individual medications. Medication groups also have a specific instance, e.g. "ACE_inhibitors" or "Nitrates" or "Angiotensin_II_receptor_blockers". Medications and medications groups contain about a dozen important properties, such as several dosage properties: "InitiatingDose", "TargetDose", "MaximumRecommendedDailyDose", "MaintenanceDose" etc.; "SideEffect", "Indicated", "Contraindicated". These properties link medications with signs and symptoms, diagnosis and other medications.

"Other_medication_groups" include those groups of medications and individual medications not directly used in the treatment of heart failure, but rather in the treatment of the most common comorbidities.

There is a total of 30 medication groups and 78 individual medications in the ontology.

6.3.4. Class "Testing"

Finally, we consider root class "Testing". It contains a thorough test list spanned through four classes: "Echocardiography_tests", "Electrocardiography_tests", "Hematology_and_biochemistry_tests" and "Other_tests". There are three individual tests under "Echocardiography_tests": "Doppler_echocardiography", "Stress_echocardiography" and "Transoesophageal_echocardiography". There are two electrocardiography based tests: "Electrocardiogram_at_rest" (12-lead) and "Holter_electrocardiography_24_hour". There are 25 hematology and biochemistry tests, for example: "C-reactive_protein_test", "Complete_blood_count", "Leukocytes", "Lipid_panel", "S-glucose" etc. "Other_tests" include 15 tests, e.g. "Cardiac_MRI", "Cardiovascular_monitoring", "Chest_CT", "6_minute_walk_test", "Thoracic_radiography". Each test has its measurements specified in separate class. All of these classes are placed in the class "Test_measurements", under "Testing". Test results relevant for inference of some disorders are placed in the separate class "Relevant_test_results", which enumerates 57 instances. For example: "Cardiothoracic_ratio_greater_than_0.5", "BNP_value_higher_than_100_pg_per_ml", "E_A_ratio_less_than_1" etc.

There also exists a separate class used to specify the normal values, most often in mg/dl or mmol/l, but also in other measure units.

7. Procedural knowledge in the heart failure domain

The HF ontology presents the detailed taxonomic overview of complete heart failure domain including relevant relations among concepts. It represents *descriptive knowledge* about the domain. Platform should also be able to perform some actions, typically in the form of suggestions for patients and medical personnel. The knowledge representing sufficient and necessary conditions that some actions can be done is the so called *procedural knowledge.* Descriptive and procedural knowledge together present the knowledge base of the HF platform.

Production rules in a form "IF *some condition is true* THEN *make some action*" are a widely used approach for the presentation of procedural knowledge. Their advantages are natural interpretation by humans and modularity during construction. It is also relevant that production rules are also a formal way of presenting knowledge and in this way a good starting point for practical realization of the decision support system. For the integration of descriptive and procedural knowledge it is important that production rules can use only the concepts defined in the HF ontology.

At the knowledge presentation level it is very important that production rules can be easily understood and corrected by medical experts. In this way the major advantage of presenting procedural knowledge in the form of production rules is that they present formal enough way to present knowledge that can be used by the platform and that at the same time medical experts can easily control the expected performance of the platform. The correction of rules or adding them should only be performed by the authorized medical personnel.

The HF procedural knowledge has been divided into 10 functional subtasks in order to enable easier human control of the completeness and consistency conditions. They are:

1. HF diagnosis
2. Alternative or additional diagnosis
3. Heart failure severity assessment – specifying NYHA class for patient, which is required for general patient treatment approach
4. HF general treatment process based on severity assessment
5. HF medications contraindications, adverse effects & additional treatment rules
6. Prognosis estimation for HF patients
7. Non-pharmacological management and recommendations
8. Specific medication prescription and dosage
9. Acute decompensation of congestive heart failure
10. Heart failure cause and CAD risk factors

The complete lists of rules in these subtasks are presented in Appendix A.

7.1. Soft computing and production rules

Intelligent medical applications require the ability to also work with imprecise or only partially true data. The goal is to ensure robustness and efficiency of the decision making process in a real world environment. Soft computing techniques including fuzzy systems and probabilistic reasoning can be used to solve these problems. These techniques typically lead to relative complex systems whose performance and final decisions are rather difficult to predict. For the platform we have decided to use solve the problem by a) a mixture of deterministic production rules with fuzzy output values (consequences) and b) complex but deterministic computation of some input values that mimic fuzzy inputs.

7.1.1. Fuzzy consequences form deterministic rules

The approach means that we have deterministic rules, deterministic rule inputs, and deterministic outputs which may have different, but in advance predefined levels of reliability or probability.

An example of the deterministic rule with fuzzy consequence is that *Heart failure is possible* IF *Patient has either heart failure signs or heart failure symptoms* AND *cardiothoracic ratio* > *0.5*. Such rule has precisely defined conditions but the level of the reliability of the consequence is rather low. Higher level of reliability is the rule with the consequence *Heart failure is probable,* while the highest level is that *Heart failure diagnosis is suggested.*

The advantage of this approach is that decision making process is deterministic and consequently relatively simple. The application of this approach is possible for the HF platform because decisions are directed to humans who must decide upon their acceptability and they are not automatically executed. Medical doctors are the only ones who can confirm and follow these suggestions. In this framework we do not have closed loop decisions and suggestions with a predefined level of reliability are completely acceptable. Moreover, they are easier to interpret by humans than the numerical values produced by probabilistic reasoning and completely fuzzy systems.

In addition to different levels of the diagnosis reliability in which we have four levels (suggested, probable, possible, unlikely), we use fuzzy conclusions for the prognosis (good, worse, very poor), for medication recommendation (suggested, consider) etc.

7.1.2. Computation of complex patient descriptors

Some deterministic rules require inputs that describe patient status that are difficult to define by absolute values. Such inputs can be described as fuzzy because the same value can in different situations have different meaning. Good examples are all values that should be interpreted relative to some other, current or previous, patient characteristics and measured values.

We avoid using fuzzy values by implementing complex computation in the process of preparing the inputs for decision making. This means that in this computation we mast take into account, besides basic patient characteristic also all other properties that significantly determine its previous or current status relevant for the interpretation of the basic characteristic. For example, for the input *significant arterial drop* we have to look into the complete patient history, compute mean blood pressure values, and then based on the current value that is more than 30 mmHg lower conclude on significant arterial drop.

The computation of complex descriptors can be effectively realized inside the factual knowledge building block. It has access to the complete set of patient data and this enables that all data necessary for complex computations can be acquired. If some data can not be found in the patient record then the final patient descriptor will have unknown value and this will prevent the respective rule to fire. For the sake of decision reliability, the rules should have also simple security cut-off points, like present systolic blood pressure below 90 mmHg in the previous example that will fire also if data necessary for complex computations are not available.

The main disadvantage of this approach is that rather complex computations relevant for the decision making process are built into fixed programmed logic with the consequence that they can not be changed easily. The advantages are simplicity of procedural knowledge and the reliability of the DSS process.

7.2. Rule set transformation and expansion

The rules presented in Appendix A are in the form and content appropriate for human understanding. Their primary role is to interface with medical experts with intention to enumerate actions (suggestions) that are expected from the platform and also to define conditions that should be fulfilled for these actions. This form of knowledge, although very precise, needs some transformations and extensions before it can be effectively implemented.

The goal of transformations is to ensure consistency of the resulting actions. For example, it is unacceptable to allow execution of the rule for probable HF diagnosis when some other rules with more reliable HF diagnosis are applicable. It practically means that we must include additional conditions which will ensure that some rules cannot be satisfied when some other rules are satisfied. Inclusion of such conditions is not a trivial task because it requires in-depth semantic understanding of the relations among medical concepts combined with the understanding of the decision making process, but it can be formally treated as a technical problem of knowledge representation for decision support tasks.

Additionally, the rules representing medical procedural knowledge in some cases implicitly allow conclusions that are not explicitly intended. For example, the rule defining the diagnosis of the *Systolic heart failure* IF *Patient has either heart failure signs or heart failure symptoms* AND *echocardiogram is abnormal* has intention to define the diagnosis. But it can be also interpreted in a way that it suggests performing echocardiography if the patient has heart failure signs and symptoms and if the echocardiography has not been done yet. Such sort of a reasoning cannot be expected from any automated reasoning system and if we want this type of conclusions, it should be explicitly coded in the form of additional rules.

The process of rule transformation and extension is illustrated in the Appendix A11, which presents the set of rules for the HF diagnosis built from the set of rules presented in the Appendix A1. Added rules in the Appendix A11 are marked by markers A1-A13. The extended set includes in total 22 rules in contrast to originally specified 9 rules. This set explicitly defines also negative diagnosis as the situation when the performed tests did not confirm positive diagnosis. If both positive and negative diagnoses are missing, then additional tests can be suggested and diagnosing process of lower reliability is acceptable.

Closed world of conclusions

The goal in given set of rules is to decide for the first visiting patient whether he should be diagnosed heart failure condition. In order to define the context of the problem, the conclusion set is closed to total of three states: positive (HF confirmed), negative (HF declined) and undefined (decision is unreachable - maybe additional information is required).

However, we can see that none of the rules makes the decision about the negative heart failure diagnosis (or undefined). Rule 4 states the only case in which diastolic HF is positive. However, by knowing that knowledge has been gathered by interviews (human context), we can assume that this is the one and the only case in which positive diastolic HF is diagnosed. In that manner, rule 4 inherently denotes that if patient has no signs or symptoms, than the diastolic heart failure diagnosis is decided to be negative. Therefore, new rule can be added to the rule base.

Rules 3 and 8 make the conclusion about the systolic heart failure based on the different causes. This situation is somewhat different than the first one, since there are two distinct rules specifying the same problem. It also can be assumed here that these are the only rules about this problem, and only if both of them are not fulfilled, the diagnosis is negative (with careful analysis of the situations where the diagnosis is undefined).

Closed world of facts

The similar situation occurs with defining facts, where all possible fact states must be denoted. E.g., when live facts about LVEF are extracted from the patient record, the rule condition has to make the distinction between the LVEF that is lower than 45, LVEF that is not lower than 45 and the missing record about the LVEF in the patient record. In that case, ambiguous situations might occur, for example in the rule for diagnosing systolic heart failure (rule 3). The echocardiography test might not have been taken yet or the LVEF value might have not been entered in the patient record. Still, the rule context must be stated for situations where systolic heart failure is decided to be: 1) positive (all information is present in patient record), 2) negative (also all required information is present) and 3) undefined (some data in the record is not yet available).

Informational gain

In cases when given rule has high importance in reaching the final decision in the system, and some data required in that rule is not yet available, it would mean a great deal if system could give advices for gathering the required data. In rule 3 that would mean that if patient has HF signs or symptoms, system should advise for an echocardiography test to be performed. This situation highly resembles the automated backward chaining procedure, with the difference that backward chaining mechanism cannot distinguish the importance of facts as humans do just by understanding the rule context.

Sequential vs. parallel interpretation

In the similar manner, rule 8 inherently suggests taking three tests: ECG, X-ray and BNP tests (if any of them has not yet been performed). By analyzing the informational gain, all of the three tests mentioned above have the same informational value (since they are connected with the AND condition). That means that all of them must be taken in order to reach positive final diagnosis. The system should decide whether to advise taking all three tests, disregarding the order of tests, or to advise sequential execution using the order as stated in the rule. In a human context, in practice tests are taken sequentially (ECG test is advised to be taken first since it is the simplest and cheapest test). However, in a decision support system, one must be careful with the sequential advising since there might occur situations where ECG test is temporarily unavailable, which could put the system on hold while at the same time X-ray test may indicate that systolic heart failure diagnosis is negative.

Rule priority

Some knowledge items might have higher priority than others. E.g. rules 5-7 state prognosis (chances) of diagnosing heart failure based on given facts. However, in a human context, it is clear that the prognosis is superfluous if the patient has already reached his final diagnosis. It is unnecessary to state that the patient probably has heart failure if other tests already indicate that it is clear that the heart failure diagnosis is positive (or negative). In that context rules 5-7 (prognosis) do not take course if rules 3, 4 and 8 (diagnosis) have already made the final decision. This can be accomplished by additional conditions in rules 5-7 to hold the prognosis if the diagnosis has been reached.

Knowledge consistency

Acquired knowledge might contain items that in given situation lead to inconsistent conclusions. The inconsistency might occur on a human level, e.g. concluding that the heart failure is both unlikely and probable at the same time (if ECG is normal and X-ray is abnormal). Inconsistency on a machine level might occur if two rules at the same time proclaim some fact to be true and not true.

Machine level inconsistencies might be dangerous for system stability while human level inconsistencies lead to semantically opposed conclusions.

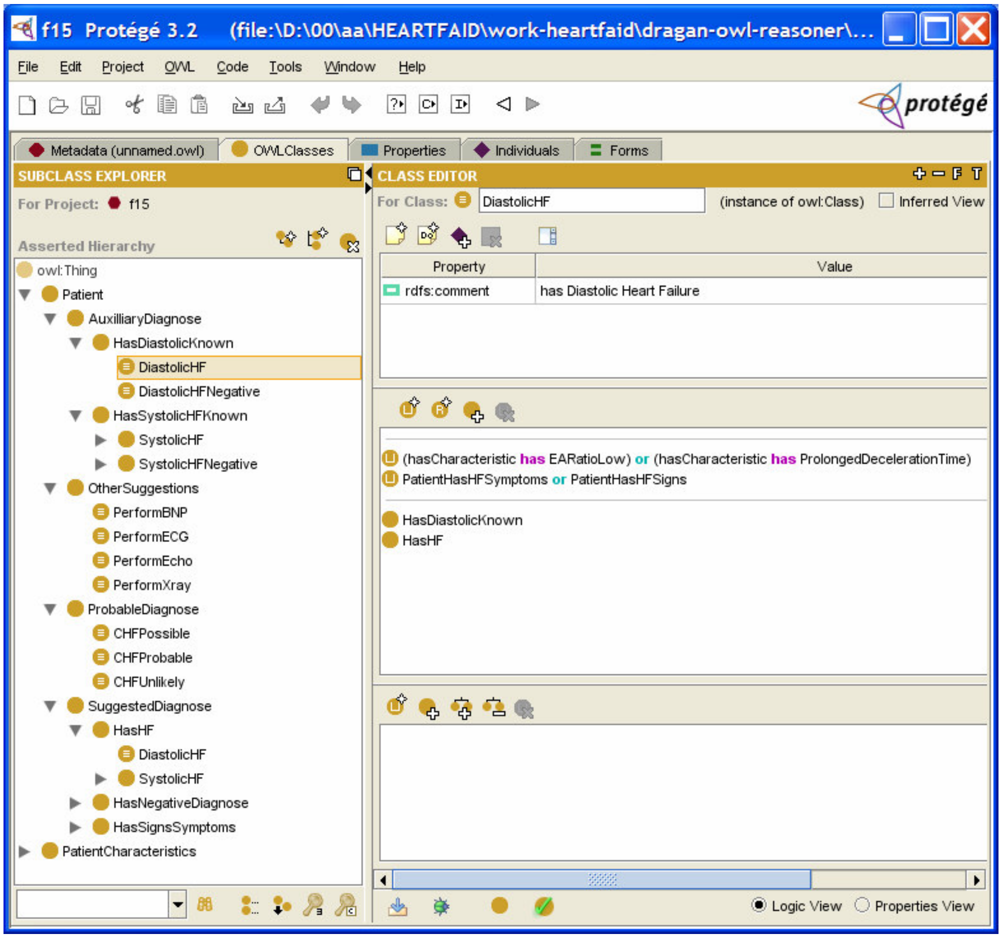
Knowledge completeness

It is clear that a set of acquired rules can never cover all possible situations that could occur in a real world. Some unexpected situations may occur outside of system's knowledge base boundaries, even though human experts could handle given situation without problems. For example, given set of rules does not indicate what happens if ECG test diagnoses that the heart failure condition is negative while the Echo test diagnoses that the heart failure condition is positive (rules 3 and 8). It is clear to human expert that Echo test is more confident than ECG test and in that case ECG test can be disregarded, but this knowledge is not encoded in the knowledge base in any way. Once that knowledge item is noticed, it can also be added into the knowledge base.

7.3. Ontological representation of procedural knowledge

Presentation of HF procedural knowledge in the form of production rules does not mean that their practical realization for the decision support purposes must be in the same form. It is true that these rules may be used to build a rule based expert system, but these rules can also be used for representing procedural knowledge in other forms. For the HF platform, the integration with descriptive knowledge presented in the HF ontology is relevant. In this situation, an appropriate form for procedural knowledge is SWRL (Semantic Web Rule Language), which is a logical extension of the OWL (Web Ontology Language).

But there is also another possibility. The unique property of OWL is the ability to represent logical operations between classes and between classes and individuals using so called *concept constructors*. It enables that logical relations contained in production rules can be presented in the ontological form. The result is the ontology that contains both descriptive and procedural knowledge. The advantage of the approach is a tight connection and conceptualized representation of complete domain knowledge which may potentially lead to a more intuitive representation of medical knowledge. In the future this can also enable web based distributed decision support, but this is not relevant for the current platform realization.



Property	Value
rdfs:comment	has Diastolic Heart Failure

(hasCharacteristic has EARatioLow) or (hasCharacteristic has ProlongedDecelerationTime)
PatientHasHFSymptoms or PatientHasHFSigns

HasDiastolicKnown
HasHF

Figure 7-1. Procedural knowledge integrated into the HF ontology. In the right part is a definition of both the necessary and the sufficient condition for the class "Diastolic HF" marked in the left part of the figure.

The procedural knowledge is most commonly presented by the production rules (classic expert system). The difference between the logic semantics in OWL and the production rules is:

- Knowledge monotony – the knowledge described using the OWL is monotonic, which means that there is no possibility to withdraw facts from the knowledge base once they have been stated. Production rules have the possibility to constantly modify facts and retract the previous statements.
- Open/closed world semantics – the reasoning process in the OWL falls under the open world semantics which understands that the knowledge base in every moment might be incompletely defined, i.e. some statements in the knowledge base might be missing. In the production rules system the reasoning is drawn under the closed world semantics, which assumes that the knowledge in the knowledge base is complete. One of the crucial features of the closed world semantics is the so called "negation-as-

failure”, which concludes that given statement is *false* if it is not currently stated to be *true* within the knowledge base.

- Function calls – the production rules systems commonly support the usage of the basic mathematical operators and user defined functions on the data in the knowledge base. The OWL does not support this kind of operations. The *numerical restrictions* in OWL may define the cardinality of a given object property, but do not support the basic comparison operators (lower-than, greater-than, equal) on the data-type properties (integer, float ...).
- Consistency check – the OWL semantics give special attention to the knowledge base consistency, giving alert messages to the user in case when inconsistencies within the knowledge base occur. The production semantics expect from a user to watch over the consistency himself (manually) in the process of knowledge base creating.

The merge of the description logics and the production rules in a *faithful* manner is currently a hot research topic [1,2] (*faithful* stands for the property that presence of one component does not affect the functionality of other, and vice-versa; when one component is not used, the other is fully functional).

Figure 7-1 shows an ontology fragment that includes the procedural rules defined as concepts using the OWL *concept constructors*. The concept *DiastolicHF* defines all the patients that should be diagnosed the diastolic heart failure condition. All concepts are organized in a concept hierarchy, which enables better organization of procedural rules and greater human-readability. The descriptive knowledge and the procedural knowledge are integrated into a single component presented as OWL ontology.

By integrating all of the ten sets of production rules, we have built the *procedural HF ontology*. It is different than the descriptive HF ontology because it has two root classes: "Patient" and "Patient_caracteristics". The "Patient_characteristics" class contains descriptive knowledge necessary for logical relations in production rules. All of its subclasses and individuals, including the class hierarchy, are based on and can be thought of as a subset of the HF ontology. Theoretically the complete descriptive HF ontology could be integrated here but this was not done because of reasoning efficency. So the "Patient_characteristics" class contains only descriptive knowledge necessary for reasoning with current version of procedural knowledge.

The class "Patient" contains the complete procedural knowledge. In order to be compatibile with production rules organization it has ten subclasses, each of them representing one rule set shown in Appendix A. These classes are: "Patient_diagnosis", "Patient_alternative_diagnosis", "Patient_severity_assessment", "Patient_general_treatment", "Patient_medical_contraindication", "Patient_prognosis", "Patient_non_pharmacological_management", "Patient_specific_medication", "Patient_decompensation" and "Patient_HF_cause". They are further divided into many subclasses. Every class with no subclasses has necessary and sufficient conditions defined, and this is where procedural knowledge is stored.

Each of the class definitions can be found as a rule in one of the rule sets. Fulfilling conditions for being in the class is perceived as a suggestion for a particular patient (e.g. class Patient_Severity_assessment has subclass NYHA_IV. Conditions for this class are: patient has dyspnoea, fatigue or palpitations at rest and patient has heart failure. The Patient with these characteristics fulfils the conditions for being in this class).

Rule sets have been built as separate entities and putting them together into a unique corpus requires some modifications of the rules. This is because suggestions acquired in one rule set can be part of the information needed for a rule in another rule set. For example, the class "Anemia_cause", which describes patients for whom anemia can be the cause for HF is an indirect subclass of "Patient_HF_cause". It has the following necessary and sufficient conditions: heart failure and anemia. The information about patient having heart failure can be collected from CRF, but it can also be suggested by the reasoning process if the conditions for appropriate subclasses of the class "Patient_diagnosis" are fulfilled. Therefore, it is necessary to add these new conditions into the original rule. In the same way the information about patient having anemia can be collected from CRF, but it can also be suggested because of fulfilling conditions for the class "Anemia_diagnosed". This class is an indirect subclass of the class "Patient_alternative_diagnosis". It is necessary to include this cognition as well. The rule is now extended and necessary and the sufficient conditions are: heart failure (imported from CRF or given as a suggestion because of fulfilling conditions for HF diagnosis), and anemia (imported from CRF or given as a suggestion because the patient fulfils conditions for the diagnosis of anemia).

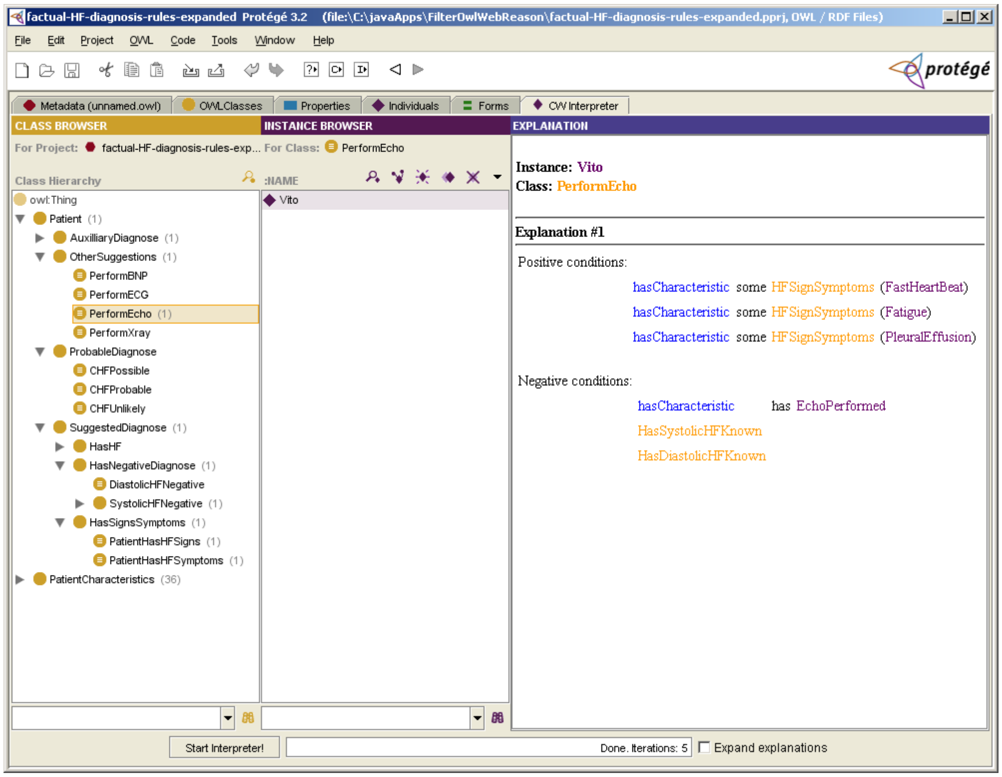
7.3.1. Interpreter for OWL procedural knowledge

We have developed a java-based OWL interpreter, introducing the "negation-as-failure" into the *instance checking* of the OWL reasoning process. We have also developed a Protégé plug-in which integrates the interpreter into the Protégé with a simple user interface. In this way, the interpreter is introduced directly into the knowledge base building facility, which eases the process of building and maintaining the knowledge base.

The introduction of the "negation-as-failure" into the interpreter's instance checking has brought the following consequences:

- Getting a step closer to the closed-world semantics which increases the simplicity and the human-readability of the rules.
- The knowledge base becomes non-monotonous; the facts can be retracted from the database.
- Possible inconsistency; the author of the knowledge base has to check for the knowledge base consistency manually.

The interpreter also provides an *explanation facility* which gives an explanation to every conclusion that has been made.



Instance: Vito
Class: PerformEcho

Explanation #1

Positive conditions:

hasCharacteristic some HFSignSymptoms (FastHeartBeat)
hasCharacteristic some HFSignSymptoms (Fatigue)
hasCharacteristic some HFSignSymptoms (PleuralEffusion)

Negative conditions:

hasCharacteristic has EchoPerformed
HasSystolicHFKnown
HasDiastolicHFKnown

Done. Iterations: 5

Figure 7-2. Explanation facility integrated in the Protégé environment.

Figure 7-2 shows the Protégé tab that integrates the interpreter and the explanation facility into the Protégé environment. It states (on the right side of the figure) that the instance *Vito* is in class “PerformEcho” because its property *hasCharacteristic* contains some instances of the class “HFSignSymptoms”, but not the *EchoPerformed* instance, while at the same time *Vito* is not a member of “HasSystolicHFKnown” nor “HasDiastolicHFKnown” classes.

7.4. Bibliography

[1] Motik, B., Sattler, U., Studer, R. (2005): *Query Answering for OWL-DL with rules*. Journal of Web Semantics, 3(1):41-60.

[2] Motik, B., Rosati, R., (2007): *A Faithful Integration of Description Logics with Logic Programming*. In Proceedings of the 20th International Joint Conference on Artificial Intelligence (IJCAI 2007) 477–482.

8. Integration of knowledge representation and patient data

The knowledge representation techniques are typically focused on the quality and wideness of knowledge representing, but at the same time they are wittingly disregarding the technical details that should make the described knowledge running and performing the desired actions. Generally, the technicalities of providing the knowledge to become usable are considered to fall into the non-intelligent part of the platform, as opposing to the knowledge itself. Still, this component of the system inevitably contains an amount of expert knowledge which interfaces the gathered knowledge to the systems resources. As a consequence, the knowledge integrating component of the system has to be easily manageable component that holds its core information both human and machine understandable and at the same time easily configurable and maintainable.

Platform's knowledge is encoded as a set of ontologies that contains the domain description (ontology concepts and relations) and procedural knowledge (description logics and productive rules). Based on a given situation, procedural knowledge recognizes the actions that are to be performed. The term "situation" mainly refers to the current state of the patient that is currently considered, but may also include some external parameters that are used in process of decision making. The system's "knowledge integrating component" has a duty to gather all the available facts from the environment and deliver it to the reasoning part in a form which the reasoner understands and can work with. The factual knowledge required by the procedural knowledge part in general comes down to ontology instances and relations among them (e.g. patient "Vito" has heart failure sign "Ortophnea"). In OWL description logics, general knowledge about a domain is referenced by the term T-box (where "T" stands for *terminology* or *taxonomy*), while the factual knowledge is referenced by the term A-box (where "A" stands for *assertion*) [10].

As within the HEARTFAID project, the platform's knowledge is configurable and therefore versatile; each modification within the knowledge base might necessitate modifications within the integration component. This setting of the platform demands for an integrating component to be highly manageable as well. Therefore, the changes within an integration component should be straightforward and prompt, considering the frequency of changes within the knowledge base and within the integration component itself.

The facts about the current medical situation are extracted mainly from the patient record, since the majority of the data required for decision making is held within it. The other sources of data might be platform messages that are delivered to the decision support system (e.g. event that contains alarming information on patient's sudden blood pressure lowering). The patient's records are commonly realized as relational databases that are, in general, institution specific. The HEARTFAID platform should be usable in various institutions, and therefore adaptable to various databases, various data structures and various available patient data. Fulfilling these constraints upon the HEARTFAID integration component enhances the platform with greater flexibility.

8.1. The environment of integration component

The primary task of the integration component is to deliver the facts about current patient status (factual knowledge) to the reasoning engine in a suitable form. While performing that task, integration component is closely communicating with:

- Platforms middleware; to reach the database (patient record) and to receive messages about the events in the platform;
- Knowledge base (ontology); to deliver the factual knowledge extracted from patient data;

The data in patient record is commonly gathered through doctor/patient interaction and in some cases automatically by running medical tests. All the relevant patient data should be stored in a patient record (data base) in order to become available for decision support process.

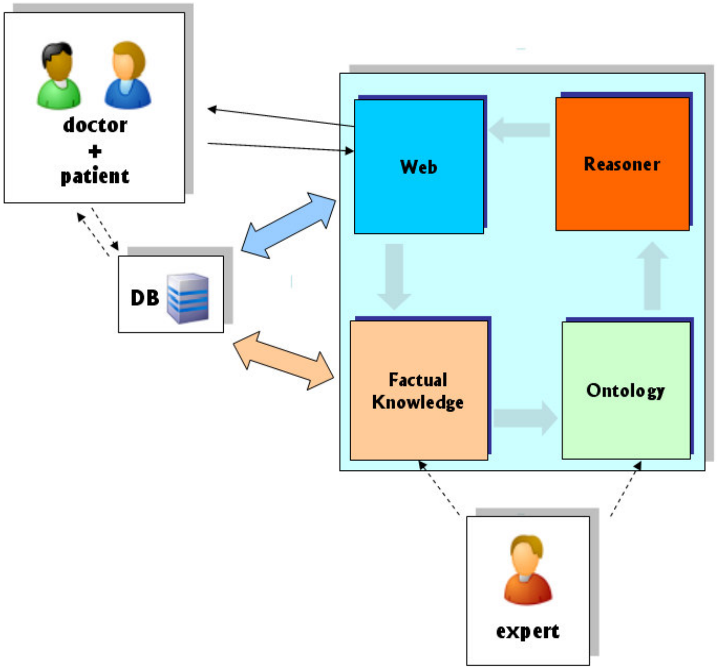


Figure 8-1. Example scenario environment of integration component (factual knowledge component)

Figure 8-1 presents the example scenario in which the knowledge integration component operates. Within this scenario the web interface (part of the end-user application and services) has a role of communication with the final user; factual knowledge component (part of the DSS) has a role of loading the factual data to the ontology; ontology (part of the knowledge base) has a role of presenting complete domain knowledge; and reasoner (part of the DSS) has a role of making the decisions based on the gathered knowledge.

The HEARTFAID platform is available to the medical practitioner (doctor) as a web service. The events performed on a web interface initiate the process of loading the factual knowledge from the patient's record (database) into the ontology. The enriched ontology is in that manner prepared for the reasoning process carried out in a reasoner component. The conclusions made in a reasoning process are served to the medical practitioner through the user interface. That closes the reasoning cycle and the system waits for the other possible events.

The role of the medical expert in such an environment is to provide the expert knowledge needed within the knowledge base. In the process of platform creation the rather frequent interventions in the knowledge base from the medical experts are expected, but in the later phases of platform creation process it is expected that knowledge base will establish rather firm and coherent shape. In the process of manipulating the knowledge in the ontology, one has to make sure that at the same time the factual knowledge is extracted appropriately, considering the changes made.

8.2. Control flow and data flow in a decision support cycle

Figure 8-2 depicts the control flow and the data flow within the example DSS scenario. The data flow encompasses the interchange of the real patient data that was extracted from the patient record (or transformed data derivatives). The control flow encompasses the interchange of control messages between the components.

The CDSS component is initially waiting in idle state. The activity on the platform generates a request that is served through the platform middleware to the CDSS interface (adapter). The request is forwarded to the DSS control unit. The control unit initiates the integration component to fetch the relevant data from the patient database (through the middleware interface). That data is transformed and loaded into ontology and into the reasoner as a set of facts. That data is transformed and sent to the reasoner as a set of facts. The reasoning process is performed and the knowledge generated is forwarded to the reasoner conclusion interpreter. Based on that knowledge, the interpreter generates a platform action that is passed through the DSS interface to the middleware. The procedure for the DSS component is finished and so it returns to the idle state, waiting for the other possible event.

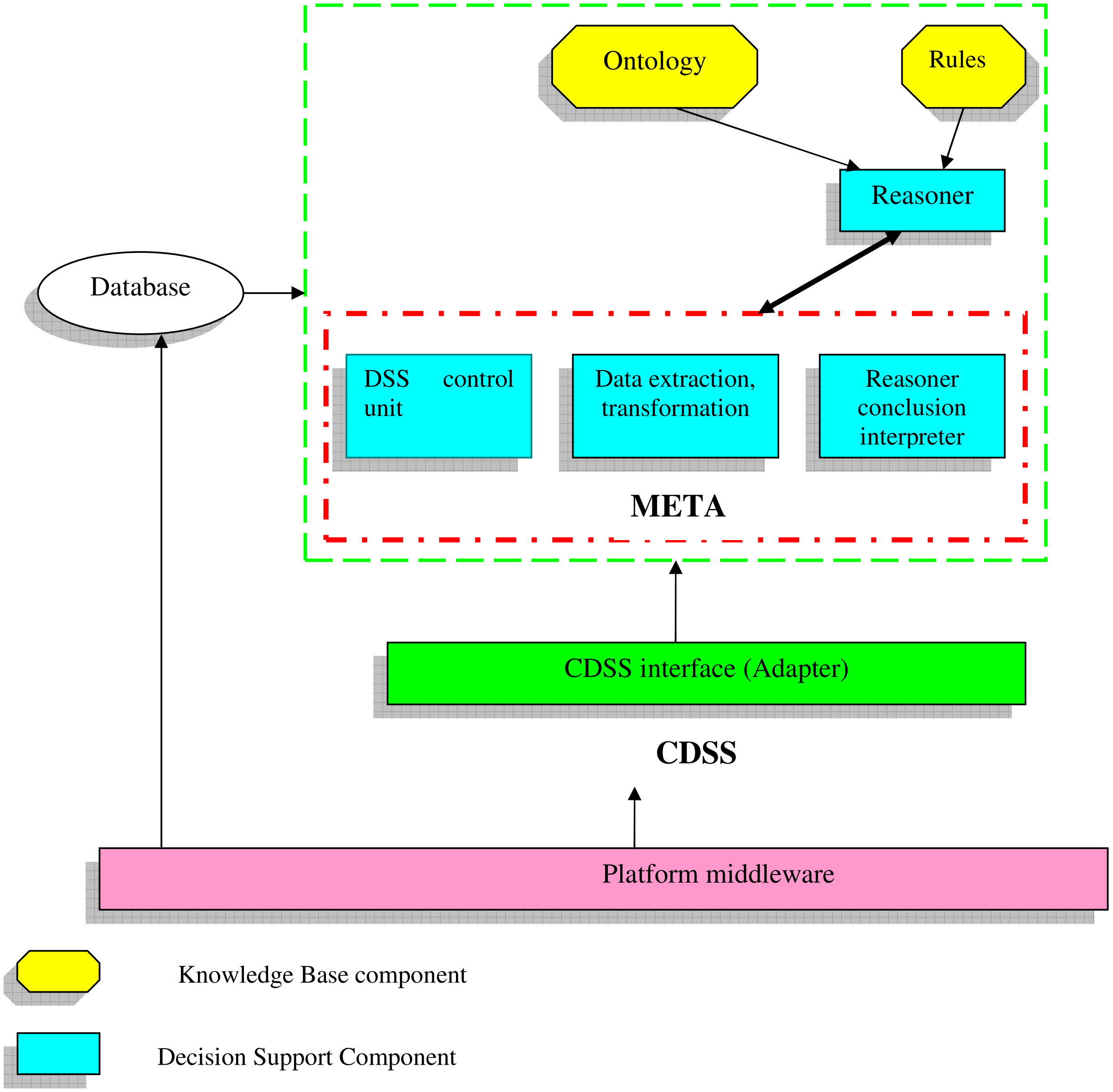


Figure 8-2. Data flow and the control flow within the example DSS scenario.

8.3. Patient record and ontology structure

To make use of knowledge stored within the ontology, the data from the patient's record is extracted and prepared for reasoning process by composing ontology individuals and arranging the relations among them.

The process of mapping the database data to ontology individuals highly depends both on a database structure and on an ontology structure. Changes in a structure on any side almost always require changes in the mapping process.

To demonstrate the mapping process we can assume a database that is structured according to HEARTFAID Case Report Form data, and heart failure diagnosis ontology that is built upon knowledge acquired in the process of knowledge acquisition. The ontology is encoded in OWL+SWRL format using Protégé ontology editor.

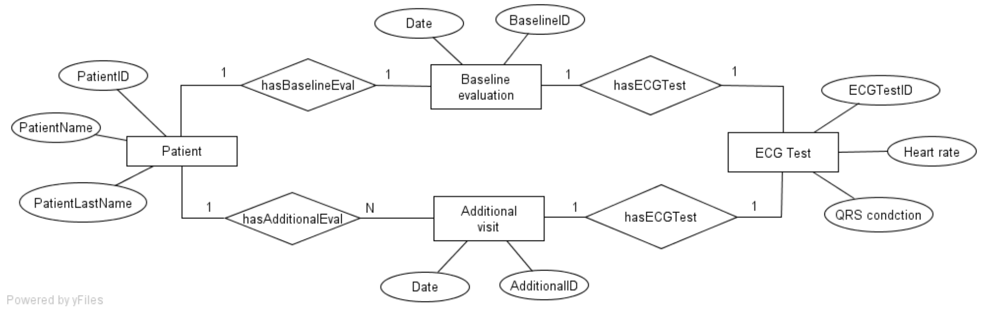
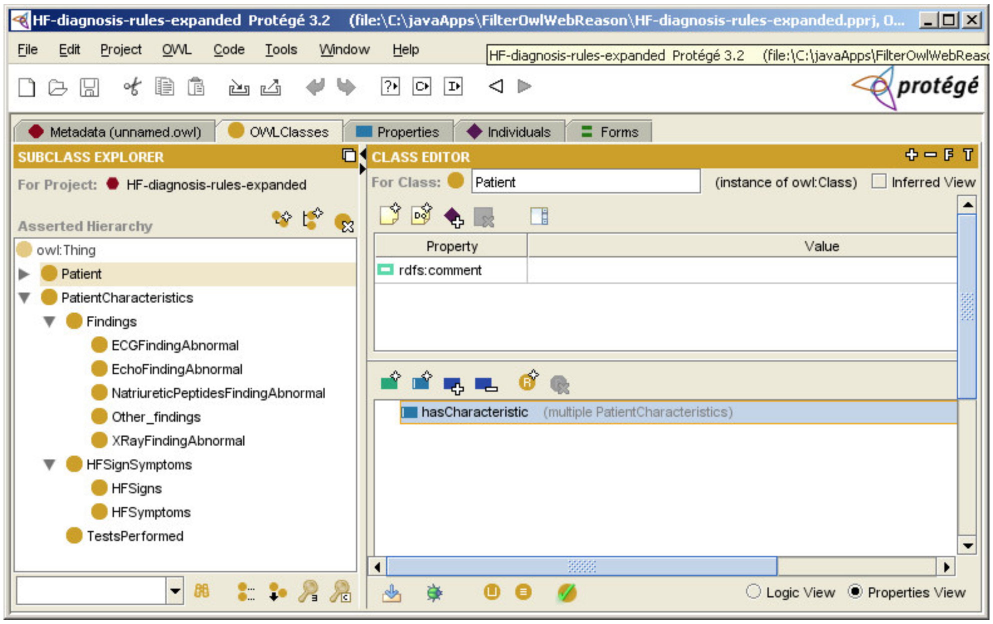


Figure 8-3. Excerpt of ER diagram of Case Report Form based Database.

Figure 8-3 shows the ER diagram of the experimental database that was built upon the Case Report Form scheme. The entity Patient holds the basic data about patients (name, last name, data of birth, etc.). Each patient has single baseline evaluation and might have any number of additional evaluations. Both additional and baseline evaluation have exactly one ECG test (12 lead). ECG test entity contains data that was gathered during each patient's ECG test, e.g. patient's heart rate and QRS conduction. Given ER excerpt is reduced only to four named entities; also, not all entity attributes have been displayed. Given database structure is generated directly from the Case Report Form entries, and is suitable for demonstration of mapping process.


HF-diagnosis-rules-expanded Protégé 3.2 (file:\C:\javaApps\FilterOwlWebReason\HF-diagnosis-rules-expanded.pprj, O...
File Edit Project OWL Code Tools Window Help
protégé
Metadata (unnamed.owl)
OWLClasses
Properties
Individuals
Forms
SUBCLASS EXPLORER
CLASS EDITOR
Logic View
Properties View

For Project: HF-diagnosis-rules-expanded

Asserted Hierarchy

- owl:Thing
  - Patient
  - PatientCharacteristics
    - Findings
      - ECGFindingAbnormal
      - EchoFindingAbnormal
      - NatriureticPeptidesFindingAbnormal
      - Other_findings
      - XRayFindingAbnormal
    - HFSignSymptoms
      - HFSigns
      - HFSymptoms
    - TestsPerformed

For Class: Patient (instance of owl:Class) □ Inferred View

Property	Value
rdfs:comment	

hasCharacteristic (multiple PatientCharacteristics)

○ Logic View ● Properties View

Figure 8-4. Excerpt from Procedural knowledge ontology.

Figure 8-4 shows the excerpt from procedural knowledge ontology. Class “Patient” holds the information about patient in property *hasCharacteristic*. The property *hasCharacteristic* allows multiple instances for the class “PatientCharacteristic”, whose subclass structure is partially shown on the left side of the Figure 8-4.

8.4. Factual knowledge extraction

The ontology developers should clearly state which ontology data items are necessary for a complete and coherent decision making. In our example ontology, one must deploy from the database all the available data about possible patient characteristics (which are listed in the class “PatientCharacteristic”).

The ontology contains knowledge about heart failure diagnosis, which is generally performed at the baseline evaluation. To simplify the problem, we will reduce the context only to the patients that are first-arriving and are just performing their baseline evaluation.

Database		Ontology	
Column	**Value**	**Property**	**Value**
heart_rhythm-sinus (boolean)	true false null	*hasCharacteristic*	HeartRhythmSinus NotHeartRhythmSinus -
heart_rhythm-other (string)	XY null	*hasHrtRhythmOth*	XY -
heart_rate_ms	XY	*hasHeartRate*	XY

(float)	null		-
conduction_PQ_ms (float)	XY null	*hasConductionPQ*	XY -
conduction_QRS_ms (float)	XY null	*hasConductionQRS*	XY -
conduction_QT_ms (float)	XY null	*hasConductionQT*	XY -
pathological_Q_waves (boolean)	true false null	*hasCharacteristic*	PathoQWaves NotPathoQWaves -
…	…	…	…

Table 8-1. Relation between the database data and the ontology data.

Table 8-1 shows the relation between the database data entries and the ontology data entries. It is evident that the database and the ontology have different structures even though they are denoting the same data entities. The first line denotes that if in the database patient *Vito* has data item *heart_rhythm-sinus* set to be *true,* inside the ontology patient *Vito* should have instance *HeartRhythmSinus* added to his property *hasCharacteristic.*

```
SELECT first_name, ecg.*
  FROM
    `ECG_12_lead_25mm_s_` AS ecg,
    `Baseline_evaluation` AS base,
    `Patient` AS patient
  WHERE
    patient.baseline_evaluation = base.Baseline_evaluation_ID
    AND base.ECG_12_lead = ecg.ECG_12_lead_25mm_s__ID
    AND patient.patient_ID= "$PatientID"

(Patient !first_name
  (hasCharacteristic ECGData ECGDataIns
    (hasHeartRate Float !heart_rate)
    (hasConduction_QRS_ms Float !conduction_QRS_ms)
  )
)
```

Figure 8-5. KBDB-ETL code for mapping ECG patient data.

Various tools able to perform mappings from Database to Ontology are available. Figure 8-5 presents a code written in KBDB-ETL tool that will cope with the above described setting.

The first part of the code is SQL query that is fetching the data from the database. The last line of the SQL query contains a constraint on the patient ID that is marked with *$PatientID*. Every pattern in a query that starts with the '$' sign is denoting the value fetched from the event message. The event on the platform that initiated the reasoning process must be aware of which patient are we currently checking. The result of a query is a data set that can be presented in a form of a table.

first_name	heart_rate	conduction_QRS_ms	...
Vito	88	130	...

Table 8-2. Data about ECG test fetched from the database.

The second part of the code from Figure 8-5 is forming the ontology instances and creating relations among them. Orange colour denotes ontology class, violet colour denotes ontology instances, blue colour denotes object properties and green colour denotes datatype properties. The patterns that are starting with the '!' sign are referencing the fetched result set (Table 8-2) by column name. That way *!first_name* is being replaced with *Vito* and *!heart_rate* is being replaced with *88*. The resulting ontology statement is shown in a Figure 8-6 and graphically presented in a Figure 8-7.

```
(Patient Vito
  (hasCharacteristic ECGData ECGDataIns
    (hasHeartRate Float 88)
    (hasConduction_QRS_ms Float 130)
  )
)
```

Figure 8-6. Data collected inside the ontology as a result of data mapping.

Instance *Vito* as its own property has instance *ECGDataIns,* which holds the information about patient's ECG test results, namely patient's heart rate and patients conduction QRS.

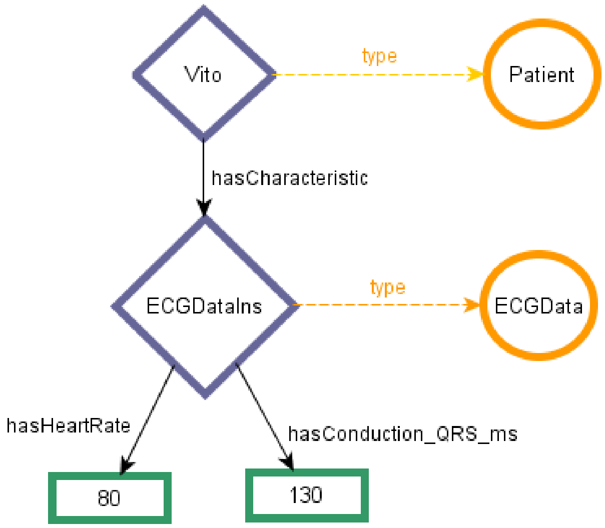


Figure 8-7. Data loaded into the ontology by the process of data mapping.

8.5. State of the art in database/ontology mapping

The described problem has been mostly addressed by the tools and methods from the area of the Semantic Web. The need for emerging the data stored in the database to the surface in the form of active and dynamic web pages is referenced by the term "deep web". The activity in that area has produced a number of various mapping tools. The mappings in the intelligent information integration area are commonly built with ad-hoc software implementations (Observer [7], Pixel [5]) that are integrated within systems and built upon the specific system requirements.

Some effort has been made in developing heuristic methods that can make the mapping process (semi)automated (KAON-Reverse [9], MAPONTO [1]). The mappings are inferred on the basis of the previously made ones (semi-automatic), or on the basis of the syntactic/structural similarities (automatic).

Three approaches have been recognized in the process of database/ontology mapping [3]:

- Generation of a completely new ontology (limitations free) based on a given DB model.
- Description of a procedure on how to fetch the data for each ontology item from the database.
- Mapping the complete existing database to the existing ontology (highest complexity).

Methods for mapping data from the database to the ontology may be compared by their expressiveness (tools may vary in ability to define ontology classes, instances and relations), ontology types supported (OWL, RDF(S), Protégé frames...), RDBMS supported (Oracle, MySQL, Informix...), price, maintenance characteristics, etc. The most well known database/ontology mapping tools are D2R MAP with it's extended version eD2R [2], R2O [3], KAON [9] and MAPONTO [1].

eD2R [2] is a powerful XML-based declarative language. It describes mappings from database to RDFS/OWL ontologies (the only supported format). XML-tags in eD2R are describing the connection to the database, SQL queries and data mapping with very high expressiveness. However, parameterised ontology and SQL statements are not implemented; they are assumed to be implemented externally. eD2R is the representative example of the second approach mentioned above.

R2O [3] is a mapping language that maps the database data to the RDF(S) or OWL schema. It is conceived to cope with complex mappings regardless of the ontology/database structure similarities and differences. The syntax of the R2O language is very expressive, but also very complex and not intended to be read or created manually by humans (GUI tools are under development). R2O provides a series of other tasks like self verification, DB integrity verification etc. R2O fits the third approach mentioned above.

KAON [9] provides a visual mapping tool to define relations between database and ontology objects. It also provides KAON-Reverse [9], a heuristic method that makes the mapping process semi-automated by analyzing database structure, including relations, attributes, attribute types, primary keys, and foreign keys/inclusion dependencies. It fits the third approach mentioned above.

Somewhat different approach is describing a database structure using the Relational.OWL schema [6] and then extracting the data by using the SPARQL ontology query language [6].

8.6. Bibliography and references

[1] An, Y., Borgida, A., Mylopoulos, J. (2004): *Refining the Semantic Mappings from Relational Tables to Ontologies*. Proceedings of 2nd International workshop on Semantic Web and Databases (SWDB'04) in conjunction with Very Large Databases (VLDB'04), p. 84-90.

[2] Barrasa, J., Corcho, O., Gomez-Perez, A. (2003): *Fund Finder Wrapper: A case study of database-to-ontology mapping*. Semantic Integration Workshop, SI-2003, p. 9-15.

[3] Barrasa, J., Gomez-Perez, A. (2006): *Upgrading Relational Legacy Data to the Semantic Web*. Proceedings of the 15th International Conference on World Wide Web (Edinburgh, Scotland). WWW '06, p. 1069-1070.

[4] de Clercq, A., Blom, J.A., Korsten, H.M., Hasman, A.(2004): *Approaches for creating computer-interpretable guidelines that facilitate decision support*. Artificial Intelligence in Medicine 31(1):1-27.

[5] Goasdoue, F., Lattes, V., Rousset, M. (2000): *The Use of CARIN Language and Algorithms for Information Integration: The PICSEL Project*. International Journal of Cooperative Information Systems, 7:127-140.

[6] Laborda, C., Conrad, S. (2005): *Relational.OWL - A Data and Schema Representation Format Based on OWL*. Second Asia-Pacific Conference on Conceptual Modelling (APCCM2005), 43:89-96.

[7] Mena, E., Kashyap, V., Sheth, A., Illarramendi, A. (2000): *OBSERVER: An Approach for Query Processing in Global Information Systems based on Interoperation across Pre-existing Ontologies*. International journal on Distributed And Parallel Databases (DAPD), 8(2):223-272.

[8] Prcela, M.; Gamberger, D.; Bogunović, N. (2007): *Developing Factual Knowledge from Medical Data by Composing Ontology Structures*. Proceedings of MIPRO 2007, Vol. III:145-150.

[9] Stojanovic, N., Stojanovic, Lj., Volz, R. (2002): *A reverse engineering approach for migrating data-intensive web sites to the Semantic Web*. Intelligent Information Processing 2002 (IIP-2002, Part of the IFIP World Computer Congress WCC2002), p. 141-154.

[10] Nardi, D., Brachman, R. J. (2003): *An Introduction to Description Logics*. Description Logic Handbook 2003: 1-40

9. Medical plans: further research topics

Medical plans are textual and visual presentations of procedures that take place after detection of some events. Events can be any type of health disorder including signs, symptoms, and diagnosis. The main characteristic of medical plans is that they are event driven and because of that they present actionable view of the medical knowledge. This actionable view is a special case of the more general procedural knowledge.

Currently, medical plans are only a middle step between experts and the guideline modelling tools which persuade the experts to clearly state the procedure they would normally perform when facing a specific problem. At the same time they enable technicians to understand it and encode it in a machine readable form [1]. They are similar to medical pathways but in contrast to medical pathways which are designed to be used by medical doctors in order to systemize and standardize their work, medical plans are designed for technical people in order to better understand medical concepts. In the HF domain we have used medical plans as an auxiliary tool for systemizing procedural knowledge development and to enable some verification of the implemented knowledge base.

The syntax of the medical plans highly resembles the traditional workflow management. The difference is that the medical plans will not be executed by machines; they are written in an almost-free graph/text form with main purpose to be fully understandable by humans. Their main characteristic is that they are event driven and their main advantage is a clear systematization of the medical procedures and interconnections among them. Additionally, their visual presentation facilitates understandability by medical experts.

For the heart failure platform, medical plans describe the disorders that can occur as events to the patient who is treated by the platform. These disorders have assigned urgency levels which correspond to the type of response needed from the medical team. For example, pulmonary edema has the highest urgency level, requiring immediate admission to the hospital. An example of a symptom that has a low urgency level is cough. Cough does not require for the patient to report to the hospital, but rather if it is persistent, he should contact his general practitioner. The urgency levels solve the problem of entering the appropriate medical plan in situations when more than one triggering event occurs. The plans of the lower urgency level can be interrupted if another event of the higher urgency level happens.

9.1. List of plans

At the moment, the heart failure system has 38 interconnected plans for signs, symptoms and diagnosis assessment and treatment and 15 plans for medications prescription and dosage. The complete list of medical plans, along with their urgency levels is given in Table 9-1 and Table 9-2.

	Plan name	Urgency level
1	Heart attack	5
2	Cerebral stroke	5
3	Ankle edema	3
4	Dizziness	2
5	Coughing	1
6	Dyspnea	3
7	Paroxysmal nocturnal dyspnea	3
8	Orthopnea	3
9	Oliguria	2
10	Rash	2
11	Involuntary weight gain	3
12	Abnormal weight loss	4
13	Angioedema	4
14	Chest pain	4
15	Fatigue	2
16	High blood pressure	3
17	Low blood pressure	3
18	Increased body temperature	3
19	Palpitations	3
20	Tachycardia	3
21	Bradycardia	4
22	Pulmonary crepitations	3
23	Pulmonary edema	5
24	Malignant hypertension	5
25	Headache	1
26	Diarrhea	2
27	Vomiting	2
28	Pleural effusion	4
29	Lightheadedness	2
30	Syncope	3
31	Abdominal pain	2
32	Sweating	3
33	Cardiogenic shock	5
34	Neurologic deficit	4
35	Inability to speak	4
36	Papilledema	4
37	Retinal hemorrhage	4
38	Atrial fibrillation	5

Table 9-1. HEARTFAID medical plans for signs, symptoms and diagnoses

	Plan name
1	Diuretics dosage
2	Diuretics plan
3	ACE inhibitor dosage
4	Beta-blocker dosage
5	Vasodilator dosage
6	Aldosterone receptor antagonists dosage
7	Fibrinolytics dosage
8	ARBs dosage
9	Inotropic agents dosage
10	Anticholinergics dosage
11	Antiarrhythmics dosage
12	Atrial fibrillation antithrombotic therapy
13	Atrial fibrillation heart rate control
14	Synus rhythm maintenance
15	Atrial fibrillation treatment

Table 9-2. Medical plans for medication prescription and dosage

It can be clearly seen from Tables 9-1 and 9-2 that not all the aspects of heart failure domain have been covered by the development of these plans. For example, plans for ACE inhibitors or beta-blockers prescription, as well as for heart failure diagnostics and general treatment, have not been designed. These topics are already covered by procedural knowledge described in Section 7.

The complete set of medical plans, in both graphical and textual form can be found on the CD that is part of this deliverable.

9.2. Building blocks for plans

Medical plans have been presented in both graphical and textual form. Usually, these two forms of presentation do not differ significantly. In some plans, however, textual representation is more thorough. This project has been using yEd graphical editor for graphical presentation of the plans. yEd tool has the option of saving the graphs in various picture (SVG, JPG, BMP, GIF) or XML formats (http://www.yworks.com/en/products_yed_about.htm).

Types of nodes and their function in plan flowcharts are defined by their shape and colour. For example, a diamond shape always represents plans, either the end of a plan (bright yellow) or the reference to another plan (light yellow) or medication dosage plan (orange). It is possible to jump from one plan to another by specifying this other plan as a node in the present plan. Full list of the so-far used shapes and colours for various medical plans functions is given in Figure 9-1.

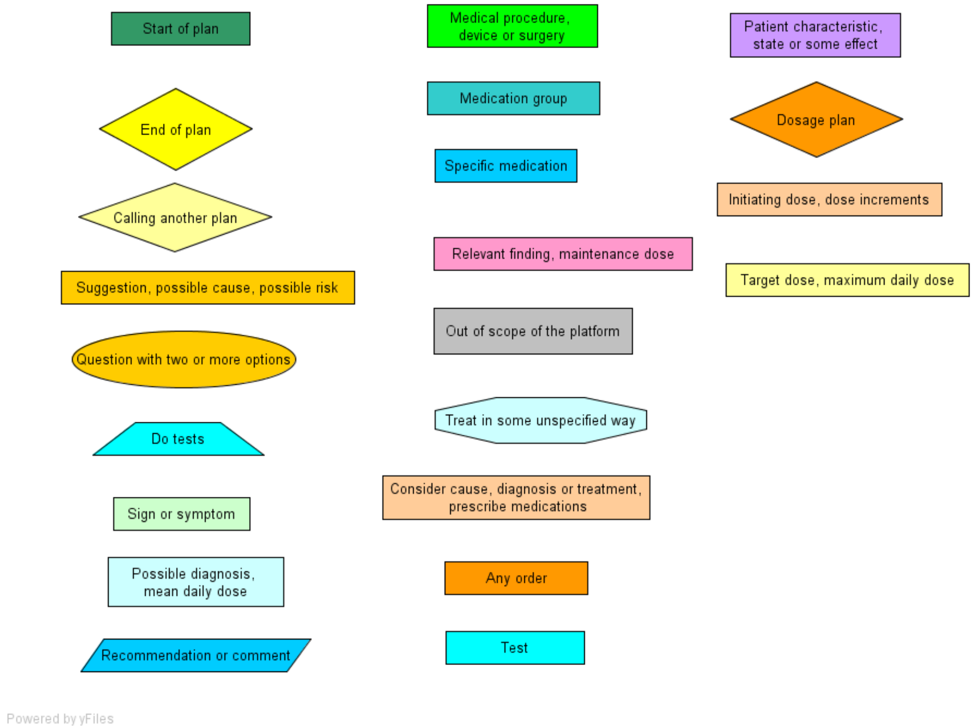

Start of plan
End of plan
Calling another plan
Suggestion, possible cause, possible risk
Question with two or more options
Do tests
Sign or symptom
Possible diagnosis, mean daily dose
Recommendation or comment
Medical procedure, device or surgery
Medication group
Specific medication
Relevant finding, maintenance dose
Out of scope of the platform
Treat in some unspecified way
Consider cause, diagnosis or treatment, prescribe medications
Any order
Test
Patient characteristic, state or some effect
Dosage plan
Initiating dose, dose increments
Target dose, maximum daily dose
Powered by yFiles

Figure 9-1. Medical plans node shapes and colours

9.3. Pulmonary edema – an example

Example of a medical plan is in Figure 9-2. It is the plan for diagnosing and handling (cardiogenic) pulmonary edema, a serious disorder caused by failing heart where the lungs are swelling and filling with fluid. Pulmonary edema has the urgency level of 5, which means it is in the class of most severe complications. It has to be treated immediately and patient has to be hospitalized.

This plan is a typical example of the level of detail with which the plans have been constructed. The plans are quite detailed; despite that they do not and can not cover all of the possible outcomes or complications of a disorder. They are meant to give general but also precise guidelines on how to treat a heart failure patient in the concrete situation. Clearly, verification and modification by the medical doctors are always welcome.

The same medical plan can be put into the textual form as well.

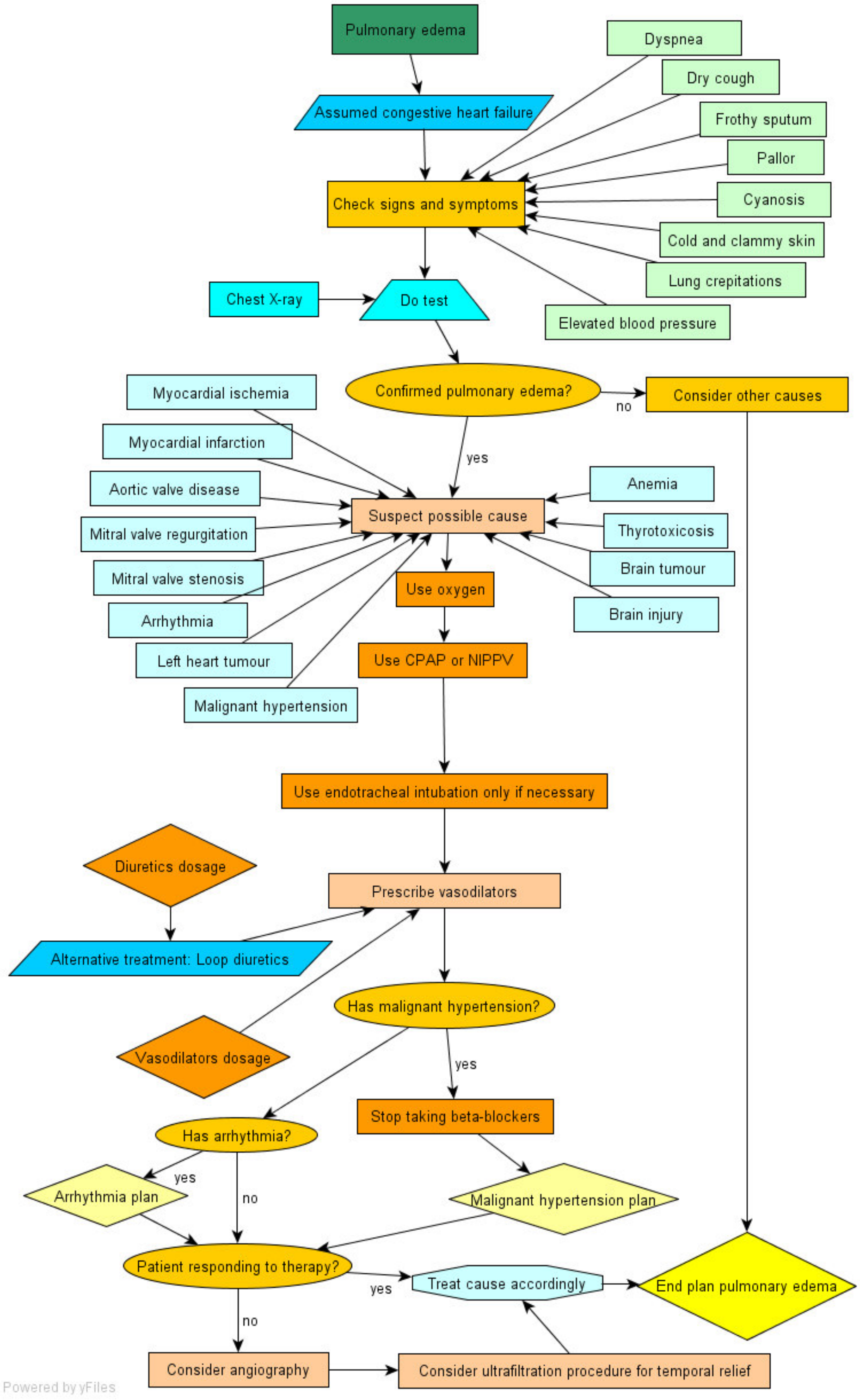


Figure 9-2. Medical plan for pulmonary edema

Plan #23
Pulmonary edema
Severity level: 5
Information for this plan has been obtained from the Acute Heart Failure Guidelines and from Congestive Heart failure Guidelines, 2005
Assume congestive heart failure

- verify by X-ray
- it is severe left heart backward failure, presenting with shortness of breath, dry cough (sometimes frothy sputum), pallor or even cyanosis, cold and clammy skin, normal or elevated blood pressure, systolic function is often preserved, also more than 50% of patients have LVEF>45%, severe respiratory distress, lung crackles (rales) and orthopnea, O2 saturation less than 90%
- can be caused by long-term hypertension, especially in older women (65+)
- possible causes are: myocardial ischemia or infarction, aortic or mitral valve disease, severe arrhythmias, left heart tumour, severe hypertension, anaemia, thyrotoxicosis, brain tumour or trauma
- treatment order is O2, then CPAP or NIPPV, then if necessary mechanical intubation, intravenous administration of antihypertensive agents (vasodilators(nitrates), nitrates are more effective than diuretics, however diuretics can also be used, do not use beta-blockers)
- consider coronary angiography if patient does not respond to treatment
- consider ultrafiltration for temporary relief

End plan #23

The usual way of designing medical plans is in close resemblance to the medical doctor's way of thinking when handling a patient. The most common procedure would be to ask for other symptoms in relation to the one the patient complains about and to do the examination and find the appropriate signs. These other signs and symptoms can either confirm the initial suspicion, or request that some tests should be taken, or completely disprove the existence of the disorder.

Usually, the next thing a medical doctor would do is to order a series of tests. These tests can also confirm the suspicions or give rise to a new possible diagnosis. The next course of action would be to prescribe the appropriate treatment or to give recommendations. The medical plans do not cover the therapeutic measures required to treat all of the disorders that can be diagnosed, because that would make the platform almost as large as the medicine itself.

All of the plans have the following things in common:

1. One node for the beginning of the plan
2. One node for the end of the plan
3. Assigned urgency level

4. Assumed heart failure of the patient, i.e. heart failure has already been diagnosed

9.4. Further research: active implementation of medical plans

In the HF project the medical plans have been developed by technical people in the phase of procedural knowledge development. The goals were to demonstrate medical domain understanding and to systemize acquired knowledge. Their significance is in the fact that they present a middle step between the experts and their expert knowledge and technical people that formalize the knowledge.

Development of medical plans opened some potentially interesting questions that might be very relevant as further research topics. The first is weather all types of useful medical procedural knowledge (or at least its major part) can be described by medical plans. If the answer is positive, then it would be interesting to think about the possibility to make medical plans executable directly without their transformation to other forms (rules, ontologies, workflows) or to try to enable their automatic conversion without human intervention [2]. Based on the work and results in the HF project, these options seem interesting because the approach based on medical plans as the first and potentially the only creative part requiring human intervention, could significantly change the traditional way of designing procedural knowledge.

9.5. Bibliography

[1] Jović, A., Prcela, M., Krstačić, G. (2007) *Medical Plans as a Middle Step in Building Heart Failure Expert System*. Proceedings of MEDICON 2007, 549-553.

[2] de Clercq, P.A., Blom, J.A., Korsten, H.H.M., Hasman, A. (2004) *Approaches for creating computer-interpretable guidelines that facilitate decision support.* Artificial Intelligence in Medicine (2004) 31, 1-27.

10. Conclusions

The deliverable presents the main results of the work related to collection, systematization, and formalization of the knowledge related to the HF domain. The main results are: descriptive HF ontology, procedural knowledge base, HF medical plans, and ontological presentation of procedural knowledge.

Although the knowledge base has been partially verified and improved by medical doctors, the current version presents technical formalization of medical guidelines and starting point for platform implementation. It can be expected that tests with prototypes will demonstrate deficiencies in the form and content of the knowledge base. By these improvements we expect to be able to collect and formalize also the tacit medical knowledge related to HF. Long and detailed experimental work with the platform is the necessary condition for the success of this process. Moreover, even in the operational life of the platform it can be expected that continuous improvements in the knowledge base will be necessary.

Effectiveness of the DSS process depends on the interplay between decision support reasoning system and the knowledge base. The development of the knowledge base presented in this deliverable concentrated mainly on capturing and formalization of medical knowledge. The following steps are transformation to a readable form for DSS and integration and verification of the complete DSS process. Last but not least is connection of the factual knowledge extraction task with the real platform data.

Appendix A: HF procedural knowledge

The HF procedural knowledge is divided into 10 relevant subtasks and for each of them a series of rules has been constructed.

A1 HF diagnosis

1.1
Patient has **heart failure signs**
IF he has at least one sign from this list

systolic blood pressure (sitting) very low <85
systolic blood pressure (sitting) very high >140
diastolic blood pressure (sitting) very low <50
diastolic blood pressure (sitting) very high > 90
heart rate very low < 40
heart rate very high > 100
jugular veins congested
S3 sound heard
S4 sound heard
breath sounds absent
crepitations present
pleural effusion present
liver enlargement present
peripheral edema (sign) present

1.2
Patient has **heart failure symptoms**
IF he has at least one symptom from this list

fatigue
dyspnea
orthopnea
paroxysmal nocturnal dyspnea
weight increase of more than 2 kg in the last month
peripheral edema (symptom)
fast heart beat (symptom)
irregular heart beat

1.3
Diagnosis: **Systolic heart failure**
IF
Patient has either heart failure signs or heart failure symptoms

AND
echocardiogram abnormal (LVEF < 45% OR left ventricular contractility decreased)

1.4
Diagnosis: **Diastolic heart failure**
IF
Patient has either heart failure signs or heart failure symptoms
AND
echocardiogram abnormal (E/A ratio < 1 OR prolonged deceleration time)

1.5
Diagnosis: **Chronic heart failure highly unlikely (90%)**
IF
Patient has either heart failure signs or heart failure symptoms
AND
ECG normal

1.6
Diagnosis: **Heart failure possible**
IF
Patient has either heart failure signs or heart failure symptoms
AND
Chest X-ray abnormal (cardiothoracic ratio > 0.5)

1.7
Diagnosis: **Heart failure probable**
IF
Patient has either heart failure signs or heart failure symptoms
AND
Chest X-ray abnormal (IF cardiothoracic ratio > 0.5 AND (pulmonary venous congestion OR pleural effusion present)

1.8
Diagnosis: **Systolic heart failure**
IF
Patient has either heart failure signs or heart failure symptoms
AND
ECG abnormal (left bundle branch block AND anterior Q waves)
AND
patient has (ischemic heart disease)
AND
Chest X-ray abnormal (cardiothoracic ratio > 0.5)
AND
natriuretic peptides abnormal (BNP > 100 pg/ml))

1.9
Diagnosis: **Heart failure**
IF
Patient has systolic heart failure
OR
patient has diastolic heart failure

A2 Alternative or additional diagnosis

2.1
Alternative OR additive diagnosis: **Anemia**
IF
Patient has low red blood cells number
AND
low hemoglobin
AND
low hematocrit

2.2
Alternative OR additive diagnosis: **Polycythemia**
IF
Patient has high red blood cells number
AND
high hemoglobin
AND
high hematocrit

2.3
Alternative OR additive diagnosis: **Diabetes mellitus**
IF
Patient has fasting serum glucose ≥ 126 mg/dl

2.4
Alternative OR additive diagnosis: **Hypercholesterolemia**
IF
Patient has total cholesterol level greater than 200 mg/dl

2.5
Detected medical problem: **Low HDL**
IF
Patient has HDL < 40 mg/dl
AND

patient is male

2.6
Detected medical problem: **Low HDL**
IF
Patient has HDL < 50 mg/dl
AND
patient is female

2.7
Detected medical problem: **High LDL**
IF
Patient has LDL > 100 mg/dl

2.8
Detected medical problem: **High triglycerides**
IF
Patient has Triglycerides > 150 mg/dl

2.9
Suggest additive OR alternative diagnosis: **Kidney failure**
IF
Patient has Glomerular Filtration Rate <90, mL/min per 1.73 m^2
OR
Patient has Serum creatinine > 133 mmol/l (1.5 mg/dl) in males or >124 mmol/l (1.4 mg/dl) in females

2.10
Suggest additive OR alternative diagnosis: **Kidney failure** OR **Medication side effect** (for ACE inhibitors AND Potassium-sparing diuretics)
IF
Patient has Serum creatinine > upper limit of local lab

2.11
Suggest additive OR alternative diagnosis: **Kidney failure** OR **Medication side effect** (for ACE inhibitors AND Potassium-sparing diuretics)
IF
Patient has Serum potassium > upper limit of local lab

2.12
Suggest additive OR alternative diagnosis: **Liver failure**
IF
Patient has AST > top limit of local lab

2.13
Suggest additive OR alternative diagnosis: **Liver failure**

IF
Patient has ALT > top limit of local lab

2.14
Additive diagnosis: **Diabetes mellitus**
IF
Patient has urine glucose present

2.15
Additive OR alternative diagnosis: **Kidney failure**
IF
Patient has urine proteins >300 mg/24 hours

2.16
Suggest alternative OR additive diagnosis: **Hypothyroidism**
IF
Patient has TSH increased

2.17
Suggest alternative OR additive diagnosis: **Hyperthyroidism**
IF
Patient has TSH decreased

2.18
Suggest alternative OR additive diagnosis: **Hyperthyroidism**
IF
Patient has FT3 increased

2.19
Suggest alternative OR additive diagnosis: **Hypothyroidism**
IF
Patient has FT3 decreased

2.20
Suggest alternative OR additive diagnosis: **Hyperthyroidism**
IF
Patient has FT4 increased

2.21
Suggest alternative OR additive diagnosis: **Hypothyroidism**
IF
Patient has FT4 decreased

A3 Heart failure severity assessment

3.1
Patient is in severity class NYHA I
IF
Patient has heart failure
AND
patient has history of heart failure signs and symptoms
AND
patient receives any treatment for heart failure
AND
patient does not have symptom (dyspnea OR fatigue OR palpitations) at ordinary physical activity)

3.2
Patient is in severity class: NYHA II
IF
Patient has heart failure
AND
patient feels comfortable at rest
AND
patient feels comfortable at less than ordinary physical activity
AND
patient has symptom(dyspnea OR fatigue OR palpitations) at ordinary physical activity

3.3
Patient is in severity class: NYHA III
IF
Patient has heart failure
AND
patient feels comfortable at rest
AND
patient has symptom (dyspnea OR fatigue OR palpitations) at less than ordinary physical activity)

3.4
Patient is in severity class: NYHA IV
IF
Patient has heart failure
AND
patient has symptom (dyspnea OR fatigue OR palpitations) at rest

A4 HF general treatment process based on severity assessment

4.1 Treatment of NYHA I
4.1.1
Continue or prescribe treatment **ACE inhibitor**
IF
Patient is in severity class NYHA I
AND
patient is tolerant of ACE inhibitor
AND
patient does not have contraindications

4.1.2
Continue or prescribe treatment: **ARB**
IF
Patient is in severity class NYHA I
AND
patient is intolerant of ACE inhibitor
AND
patient is tolerant of ARB
AND
patient does not have contraindications

4.1.3
Continue treatment OR prescribe treatment: **Beta-blocker** AND **Aldosterone receptor antagonist** AND
IF
patient is in severity class NYHA I
AND
patient had myocardial infarction
AND
patient does not have contraindications

4.1.4
Don't prescribe treatment OR reduce treatment OR stop treatment: **Diuretic**
IF
patient is in severity class NYHA I
AND
patient does not have signs or symptoms of fluid retention

4.1.5

Prescribe treatment OR continue treatment: **Cardiac glycoside**
IF
patient is in severity class NYHA I
AND
patient has atrial fibrillation
AND
patient does not have contraindications

4.2 Treatment of NYHA II
4.2.1
Prescribe treatment: **ACE inhibitor**
IF
patient is in severity class NYHA II
AND
patient is tolerant of ACE inhibitor
AND
patient does not have contraindications

4.2.2
Prescribe treatment: **ARB**
IF
patient is in severity class NYHA II
AND
patient is intolerant of ACE inhibitor
AND
patient does not have contraindications

4.2.3
Prescribe treatment OR continue treatment: **Beta-blocker**
IF
patient is in severity class NYHA II
AND
(patient takes ACE inhibitor
OR
patient takes ARB)
AND
patient does not have contraindications

4.2.4
Prescribe treatment: **Aldosterone receptor antagonist**
IF
patient is in severity class NYHA II
AND
patient had myocardial infarction

AND
patient does not have contraindications

4.2.5
Prescribe treatment: **Loop diuretic**
IF
patient is in severity class NYHA II
AND
patient has fluid retention
AND
(patient has glomerular filtration rate < 30 ml/min
OR
Thiazide is intolerated
OR
Thiazide is contraindicated)
AND
patient does not have contraindications

4.2.6
Prescribe treatment: **Thiazide**
IF
patient is in severity class NYHA II
AND
patient has fluid retention
AND
patient has glomerular filtration rate >= 30 ml/min
AND
patient does not have contraindications

4.2.7
Prescribe treatment: **Cardiac glycoside**
IF
patient is in severity class NYHA II
AND
patient has atrial fibrillation
AND
patient does not have contraindications

4.2.8
Continue treatment: **Cardiac glycoside**
IF
patient is in severity class NYHA II
AND
(patient was in severity class NYHA III
OR (patient was in severity class NYHA IV AND patient was taking Cardiac glycoside)

)
AND
patient does not have contraindications

4.2.9
Consider additional treatment: **ARB**
IF
patient is in severity class NYHA II
AND
patient takes ACE inhibitor
AND
patient has heart failure signs and symptoms
AND
patient does not have contraindications

4.3 Treatment of NYHA III

4.3.1
Prescribe OR continue treatment: **ACE inhibitor**
IF
patient is in severity class NYHA III
AND
patient is tolerant of ACE inhibitor
AND
patient does not have contraindications

4.3.2
Prescribe OR continue treatment: **ARB**
IF
patient is in severity class NYHA III
AND
patient is intolerant of ACE inhibitor
AND
patient does not have contraindications

4.3.3
Prescribe OR continue treatment: **Beta-blocker**
IF
patient is in severity class NYHA III
AND
(Patient takes ACE inhibitor
OR
Patient takes ARB)
AND

patient does not have contraindications

4.3.4
Prescribe OR continue treatment: **Aldosterone receptor antagonist**
IF
patient is in severity class NYHA III
AND
patient does not have contraindications

4.3.5
Prescribe treatment: **Loop diuretic**
IF
patient is in severity class NYHA III
AND
patient has fluid retention
AND
patient does not have contraindications

4.3.6
Increase dosage of loop diuretic
IF
patient is in severity class NYHA III
AND
(patient was in severity class NYHA II
OR
Fluid retention persists on current dosage)
AND
patient does not have contraindications

4.3.7
Prescribe treatment OR continue treatment: **Cardiac glycoside**
IF
patient is in severity class NYHA III
AND
patient has heart failure signs and symptoms
AND
(patient takes ACE inhibitor
OR
patient takes beta-blocker)
AND
patient takes diuretic
AND
patient takes spironolactone
AND
patient does not have contraindications

4.3.8
Consider additional treatment: **ARB**
IF
patient is in severity class NYHA III
AND
patient takes ACE inhibitor
AND
patient has heart failure signs and symptoms
AND
patient does not have contraindications

4.3.9
Consider treatment: **Thiazide**
IF
patient is in severity class NYHA III
AND
patient has fluid retention
AND
patient takes loop diuretic
AND
patient does not have contraindications

4.4 Treatment of NYHA IV

4.4.1
Continue treatment: **ACE inhibitor**
IF
patient is in severity class NYHA IV
AND
patient is tolerant of ACE inhibitor
AND
patient does not have contraindications

4.4.2
Continue treatment: **ARB**
IF
patient is in severity class NYHA IV
AND
patient is intolerant of ACE inhibitor
AND
patient does not have contraindications

4.4.3
Continue treatment: **Beta-blocker**
IF

patient is in severity class NYHA IV
AND
patient does not have contraindications

4.4.4
Continue treatment: **Aldosterone receptor antagonist**
IF
patient is in severity class NYHA IV
AND
patient does not have contraindications

4.4.5
Continue treatment AND increase dosage: **Loop diuretic** AND **thiazide diuretic**
IF
patient is in severity class NYHA IV
AND
patient does not have contraindications

4.4.6
Consider treatment: **Metolazone**
IF
patient is in severity class NYHA IV
AND
patient does not have contraindications

4.4.7
Continue treatment: **Cardiac glycoside**
IF
patient is in severity class NYHA IV
AND
patient does not have contraindications

4.4.8
Consider additional treatment: **ARB**
IF
patient is in severity class NYHA IV
AND
patient takes ACE inhibitor
AND
patient has heart failure signs and symptoms
AND
patient does not have contraindications

4.4.9
Consider intermittent treatment: **Inotropic agents**
IF

patient is in severity class NYHA IV
AND
patient does not have contraindications

4.4.10
Consider heart transplantation
IF
patient is in severity class NYHA IV

A5 HF medications contraindications, adverse effects & additional treatment rules

5.1 Beta-blockers
5.1.1
“Starting of beta-blocker treatment is not recommended”
IF
patient takes inotropic agents
OR
patient has fluid retention

5.1.2
“Administration of beta-blockers is contraindicated”
IF
patient has bronchial asthma
OR
patient has severe chronic obstructive pulmonary disease
OR
patient has bradycardia
OR
patient has hypotension

5.2 Aldosterone receptor antagonists
5.2.1
“Aldosterone receptor antagonists increase risk of hyperkalemia”
IF
patient takes aldosterone receptor antagonists
OR
patient takes potassium sparing diuretics

5.2.2
“Spironolactone can cause painful gynecomastia in 10% of cases”

IF
patient takes spironolactone

5.3 Diuretics

5.3.1

"Diuretics in excessive use can cause renal failure, often orthostatic hypotension, reduction of cardiac preload, which causes lower cardiac output and lower stroke volume"
IF
patient takes diuretics

5.3.2

Consider treatment: **Potassium-sparing diuretics**
IF
patient takes diuretics
AND
patient takes ACE inhibitors
AND
patient has hypokalemia
AND
patient does not take potassium-sparing diuretics

5.3.3

Consider dose increase: **Diuretics** AND Combination of treatment: **Loop diuretics and Thiazides**
IF
patient takes diuretics
AND
patient has insufficient response

5.3.4

"Recommended administration of loop diuretic twice daily"
IF
patient takes diuretics
AND
patient has persistent fluid retention

5.3.5

"Recommended monitoring by repeated measurements every 5-7 days of serum potassium and creatinine"
IF
patient takes potassium-sparing diuretics

5.3.6

“Warning: patient must take diuretics only in combination with ACE inhibitors”
IF
patient takes diuretics
AND
patient does not take ACE inhibitors

5.3.7
Reduce treatment: **Diuretics**
IF
patient takes diuretics
AND
patient is dehydrated

5.4 ACE inhibitors
5.4.1
“ACE inhibitors’ side effects are: dry cough, hypotension, renal insufficiency, hyperkalemia, angioneurotic edema and syncope”
IF
patient takes ACE inhibitors

5.4.2
Consider stopping treatment: **ACE inhibitors**
IF
patient has severe cough
AND
patient takes ACE inhibitors

5.4.3
“Administration of ACE inhibitors is contraindicated”
IF
patient does not take ACE inhibitors
AND
patient was taking ACE inhibitors
AND
patient had bilateral renal artery stenosis OR patient had angioneurotic edema

5.4.4
“Treatment initiation of ACE inhibitors should be pursued under specialist care”
IF
patient has low systolic blood pressure (<100 mmHg) OR patient has high serum creatinine (>250 umol/l))
AND

patient does not take ACE inhibitors

5.4.5
“Patient has increased risk of developing hyperkalemia”
IF
patient takes ACE inhibitors
AND
patient takes potassium-sparing diuretics
AND
patient does not have hypokalemia

5.4.6
“ACE inhibitors prescription is contraindicated”
IF
patient has high serum potassium (>5.5 umol/l)

5.5 Potassium-sparing diuretics
5.5.1
Suggest treatment: **Potassium-sparing diuretics**
IF
patient has persistent hypokalemia
AND
patient takes ACE inhibitors
AND
patient does not take potassium-sparing diuretics

5.6 Cardiac glycosides
5.6.1
Suggest treatment: **Cardiac glycosides**
IF
patient has atrial fibrillation
AND
patient does not take cardiac glycosides

5.6.2
“Cardiac glycosides are contraindicated”
IF
patient has bradycardia
OR
patient has second degree AV block
OR
patient has third degree AV block
OR

patient has sick sinus syndrome
OR
patient has carotid sinus syndrome
OR
patient has Wolff-Parkinson-White syndrome
OR
patient has hyperthrophic obstructive cardiomyopathy
OR
patient has hypokalemia
OR
patient has hyperkalemia

5.7 Vasodilator agents and nitrates
5.7.1
Suggest treatment: **Vasodilator agents** AND **Nitrates**
IF
patient has intolerance of ACE inhibitors
AND
patient has intolerance of ARBs
AND
patient does not take vasodilator agents AND nitrates

5.7.2
Consider additional treatment: **Nitrates**
IF
patient has dyspnea
AND
patient has angina pectoris
AND
patient does not take nitrates

5.7.3
Prescribe treatment: **Nitrates**
IF
patient has cardiogenic shock

5.7.4
Prescribe treatment: **Vasodilator agents**
IF
patient has pulmonary edema
AND
patient had taken oxygen
AND
patient had taken CPAP OR patient has device NIPPV

5.7.5
"Nesiritide may cause hypotension"
IF
patient takes nesiritide

5.8 Antithrombotic agents
5.8.1
Prescribe treatment: **Anticoagulants**
IF
patient has atrial fibrillation
OR
patient had thromboembolic event
OR
patient has left ventricular thrombus

5.8.2
Prescribe treatment: **Anticoagulants**
IF
patient has coronary artery disease

5.8.3
Prescribe treatment: **Asprin** OR **Anticoagulants**
IF
patient had myocardial infarction

5.8.4
"Aspirin should be avoided"
IF
patient has recurrent hospitalizations
AND
patient is in severity class NYHA III OR patient is in severity class NYHA IV

5.9 Anti-arrhythmics class III
5.9.1
Prescribe medication: **Amidarone**
IF
patient has ventricular arrhythmia
OR
patient has ventricular arrhythmia
OR

patient has atrial fibrillation
OR
patient has atrial flutter
OR
patient undergoes procedure electric countershock

5.10 Oxygen
5.10.1
Prescribe treatment: **Oxygen**
IF
patient has pulmonary edema

5.11 Revascularization
5.11.1
Consider treatment: **Revascularization**
IF
patient has coronary artery disease
AND
patient has myocardial infaction OR patient has myocardial hibernation

5.12 Left ventricular aneurysmectomy
5.12.1
Perform treatment: **Left ventricular aneurysmectomy**
IF
patient has left ventricular aneurysm

5.13 Pacing devices and defibrillators
5.13.1
Consider treatment: **Right ventricular pacing**
IF
patient has sustained bradycardia

5.13.2
Consider treatment: **Biventricular pacing** AND **Implantable cardioverter defibrillator**
IF
patient has reduced LVEF (<35%)
AND

broad QRS (QRS width > 120 ms)
AND
patient is in severity class NYHA III OR patient is in severity class NYHA IV

5.13.3
Consider treatment: **Implantable cardioverter defibrillator**
IF
patient had cardiac arrest
OR
patient has sustained ventricular tachycardia

5.13.4
Consider treatment: **Implantable cardioverter defibrillator**
IF
patient has reduced LVEF (<30-35%)
AND
patient did not have myocardial infarction in last 40 days

5.14 Heart transplantation
5.14.1
"Heart transplantation is contraindicated"
IF
patient is alcohol abuser
OR
patient is drug abuser
OR
patient has lack of co-operation
OR
patient has uncontrolled mental illness
OR
patient has cancer with remission and < 5 years follow up
OR
patient has systemic disease (with multi-organ involvement)
OR
patient has uncontrolled infection
OR
patient has renal failure (creatinine clearance < 50 ml/min OR creatinine > 250 umol/l)
OR
patient has fixed high pulmonary vascular resistance
OR
patient had recent thromboembolic complication
OR

patient has peptic ulcer
OR
patient has liver failure

5.15 Ventricular assist device and artificial heart
5.15.1
Consider treatment: **Ventricular assist device** OR **artificial heart**
IF
patient has acute severe myocarditis
OR
patient has suggested treatment heart transplantation

5.16 Ultrafiltration
5.16.1
Consider treatment: **Ultrafiltration**
IF
patient has pulmonary edema
OR
patient has peripheral edema
OR
patient is in NYHA IV

5.17 Drugs to avoid
5.17.1
"The use of these drugs is not recommended when receiving heart failure medical treatment. Discontinuing or using them with caution is highly recommended"
IF
(patient takes NSAIDs
OR
patient takes class I antiarrhythmic
OR
patient takes calcium antagonists
OR
patient takes tricyclic anti-depressants
OR
patient takes corticosteroids
OR
patient takes lithium
OR
patient takes coxibs)

AND
patient has heart failure

A6 Prognosis estimation for HF patients

Comment: The more contributing risk factors and commorbidities a patient has, the worse is his prognosis. Used levels are (good, worse, very pure).

6.1 Demographics and historical
6.1.1
Worse prognosis
IF
patient is advanced age
OR
patient has history of heart disease
OR
patient has diabetes mellitus
OR
patient has resuscitated sudden death
OR
patient is in risky ethnic group

6.2 Clinical
6.2.1
Worse prognosis
IF
patient has high heart rate
OR
patient has persistent low blood pressure
OR
patient is in severity class NYHA III
OR
patient is in severity class NYHA IV
OR
patient has involuntary weight loss
OR
patient has ventilatory rhythm and rate disturbance

6.3 Electrophysiologic
6.3.1
Worse prognosis

IF
patient has broad QRS (>120 ms)
OR
patient has low heart rate variability
OR
patient has complex ventricular rhythms
OR
patient has T wave alternans
OR
patient has decreased blood pressure variability
OR
patient has decreased baroreflex sensitivity

6.4 Functional
6.4.1
Worse prognosis
IF
patient has low VO2 max (ml/kg*min < 10-14)
OR
patient has low 6 min walking ability
OR
patient has high VE/VCO2 ratio

6.5 Blood
6.5.1
Worse prognosis
IF
patient has high serum BNP (>100 pg/ml)
OR
patient has high serum norepinephrine
OR
patient has low serum sodium
OR
patient has high serum creatinine
OR
patient has high serum bilirubine
OR
patient has anemia
OR
patient has high serum uric acid

6.6 Hemodynamic
6.6.1

Worse prognosis
IF
patient has low LVEF
OR
patient has increased LV volume
OR
patient has low cardiac index
OR
patient has high left ventricular filling pressure
OR
patient has restrictive mitral filling pattern
OR
patient has impaired right ventricular function
OR
patient has high cardiothoracic ratio

6.7 Relevant prognostic markers
6.7.1
Worse prognosis
IF
patient has low LVEF
AND
patient is in severity class NYHA I

6.7.2
Good prognosis
IF
patient has preserved LVEF
AND
patient is in severity class NYHA I

6.7.3
Worse prognosis
IF
patient has plasma volume changes over time
AND
patient has onset or worsening of mitral regurgitation

6.7.4
Good prognosis
IF
patient has normal or low BNP level (<100 pg/ml)
AND
patient is in severity class NYHA I OR patient is in severity class NYHA II

6.7.5

Very poor prognosis

IF

patient has impaired right ventricular function

AND

patient is in severity class NYHA IV

6.7.6

Very poor prognosis

IF

(patient has high serum creatinine

OR

patient has high serum bilirubine

OR

patient has hyponatremia

OR

patient has renal dysfunction

OR

patient has renal failure)

AND

patient is in severity class NYHA IV

6.7.7

Very poor prognosis

IF

patient has fixed high pulmonary vascular resistance

AND

patient is in severity class NYHA IV

6.7.8

Worse prognosis

IF

patient has decreased exercise capacity

AND

high VE/VCO2 ratio

6.7.9

Very poor prognosis

IF

patient has cardiac cachexia

6.7.10

Worse prognosis

IF

patient is in severity class NYHA I

AND
patient does not have heart failure signs or symptoms
AND
patient was not in severity class NYHA II
AND
patient was not in severity class NYHA III
AND
patient was not in severity class NYHA IV

A7 Non-pharmacological management and recommendations

7.1 Dietary measurements
7.1.1
"Patient is advised to reduce the amount of salt in the diet"
IF
(patient is in severity class NYHA III OR patient is in severity class NYHA IV)
AND
(patient has increased salt intake OR patient has hypernatremia)

7.1.2
"Patient is discouraged from taking salt substitutes with medications, because that can lead to hyperkalemia"
IF
patient has salt substitutes intake
AND
patient takes ACE inhibitors OR patient takes aldosterone receptor antagonists

7.1.3
"Fluid should be restricted to 1.5-2 litres per day"
IF
patient is in severity class NYHA III
OR
patient is in severity class NYHA IV

7.1.4
"Alcohol consumption is prohibited because of alcoholic cardiomyopathy"
IF
patient has alcoholic cardiomypathy

7.1.5
"Weight reduction is recommended."

IF
patient is obese
OR
patient is overweight

7.1.6
“Smoking is always discouraged”
IF
patient is a smoker

7.1.7
“Patient has intermediate risk of cardiac decompensation triggered by sexual activity. Recommend the use of sublingual nitrates before sexual activity. Discourage major emotional involvement”
IF
patient is in severity class NYHA II

7.1.8
“Patient has high risk of cardiac decompensation triggered by sexual activity. Recommend the use of sublingual nitrates before sexual activity. Discourage major emotional involvement”
IF
patient is in severity class NYHA III
OR
patient is in severity class NYHA IV

7.1.9
“Exercise training programmes are encouraged”
IF
patient is in severity class NYHA II
OR
patient is in severity class NYHA III)

A8 Specific medication prescription and dosage

Comment: Only the most common medications in HF treatment for each group are mentioned

8.1 ACE inhibitors
8.1.1

Consider medications: **Captopril, Enalapril, Lisinopril, Ramipril or Trandolapril**
IF
patient is suggested ACE inhibitors

8.1.2
"Gradually up-titrate from lower dose to target dose"
IF
patient is suggested ACE inhibitors

8.1.3
"Initiating dose: 6.25 mg t.i.d; Target dose: 50 mg"
IF
patient is suggested Captopril

8.1.4
"Initiating dose: 2.5 mg/day; Target dose: 10 mg b.i.d or 20 mg b.i.d"
IF
patient is suggested Enalapril

8.1.5
"Initiating dose: 2.5 mg/day; Target dose: High dose: 32.5-35 mg; low dose: 2.5-5 mg"
IF
patient is suggested Lisinopril

8.1.6
"Initiating dose: 1.5-2 mg/day; Target dose: 5 mg b.i.d"
IF
patient is suggested Ramipril

8.1.7
"Initiating dose: 1 mg/day; Target dose: 4 mg/day"
IF
patient is suggested Trandolapril

8.2 Diuretics
8.2.1
Consider medications: **Bumetanide, Furosemide, Torasemide**
IF
patient is suggested loop diuretics

8.2.2
"Initiating dose: 0.5-1.0 mg; Maximum daily dose: 5-10 mg"

IF
patient is suggested Bumetanide

8.2.3
“Initiating dose: 20-40 mg; Maximum daily dose: 250-500 mg”
IF
patient is suggested Furosemide

8.2.4
“Initiating dose: 5-10 mg; Maximum daily dose: 100-200 mg”
IF
patient is suggested Torasemide

8.2.5
Consider medications: **Bendroflumethiazide, Hydrochlorothiazide**
IF
patient is suggested thiazides

8.2.6
“Initiating dose: 2.5 mg; Target dose: 10 mg”
IF
patient is suggested Bedroflumethiazide

8.2.7
“Initiating dose: 25 mg; Target dose: 50-75 mg”
IF
patient is suggested Hydrochlorothiazide

8.2.8
Consider medications: **Amiloride, Triamterene**
IF
patient is suggested potassium-sparing diuretics

8.2.9
“Initiating dose: 2.5 mg +ACEI; 5 mg –ACEI ; Maximum daily dose: 20 mg +ACEI; 40 mg –ACEI”
IF
patient is suggested Amiloride

8.2.10
“Initiating dose: 25 mg +ACEI; 50 mg –ACEI ; Maximum daily dose: 100 mg +ACEI; 200 mg –ACEI”
IF
patient is suggested Triamterene

8.2.11

“Initiating dose: 2.5 mg; Target dose: 10 mg”
IF
patient is suggested Metolazone

8.3 Beta-blockers
8.3.1
Consider medications: **Bisoprolol, Metoprolol succinate, Carvedilol, Nebivolol**
IF
patient is suggested beta-blockers

8.3.2
“Gradually up-titrate from lower dose to target dose, doubling every 3 weeks”
IF
patient is suggested beta-blockers

8.3.3
“Initiating dose: 1.25 mg/day; Increments: 2.5, 3.75, 5, 7.5, 10 mg/day; Target dose: 10 mg/day”
IF
patient is suggested Bisoprolol

8.3.4
“Initiating dose: 12.5 or 25 mg/day; Increments: 25, 50, 100, 200 mg/day; Target dose: 200 mg/day”
IF
patient is suggested Metoprolol succinate)

8.3.5
“Initiating dose: 3.125 mg; Increments: 6.25, 12.5, 25, 50 mg/day; Target dose: 50 mg/day”
IF
patient is suggested Carvedilol

8.3.6
“Initiating dose: 1.25 mg/day; Increments: 2.5, 5, 10 mg/day; Target dose: 10 mg/day”
IF
patient is suggested Nebivolol

8.4 Aldosterone receptor antagonists
8.4.1

Consider medications: **Eplerenone, Spironolactone**
IF
patient is suggested Aldosterone receptor antagonists

8.4.2
“Initiating dose: 25 mg/day; Target dose: 50 mg/day. Increase to target dose if symptoms persist after 1 month of treatment”
IF
patient is suggested Eplerenone

8.4.3
“Initiating dose: 12.5-25 mg +ACEI; 50 mg –ACEI; Maximum recommended daily dose: 50 mg +ACEI; 100-200 mg –ACEI. Increase dose if symptoms persist after 1 month of treatment”
IF
patient is suggested Spironolactone

8.5 Angiotensin II receptor blockers (ARBs)
8.5.1
Consider medication: **Candesartan, Valsartan**
IF
patient is suggested ARBs

8.5.2
“Gradually up-titrate from lower dose to target dose”
IF
patient is suggested ARBs

8.5.3
“Initiating dose: 4 mg/day; Target dose: 32 mg/day”
IF
patient is suggested Candesartan

8.5.4
“Initiating dose: 40 mg/day; Target dose: 160 mg/day”
IF
patient is suggested Valsartan

8.6. Cardiac glycosides
8.6.1
Consider medication: **Digoxin**
IF

patient is suggested cardiac glycosides

8.6.2
"Initiating and target dose: 0.0625-0.125 mg/day"
IF
patient is suggested Digoxin

8.7 Vasodilator agents and nitrates
8.7.1
Consider medication: **Hydralazine** AND **Isosorbide dinitrate, Nesiritide**
IF
patient is suggested vasodilator agents AND nitrates

8.7.2
"Target dose: Hydralazine up to 300 mg – ACEI; Isosorbide dinitrate up to 160 mg – ACEI"
IF
patient is suggested Hydralazine AND Isosorbide dinitrate

8.7.3
"Target dose: 2 ug/kg IV bolus, 0.015-0.03 ug/kg/min continuous infusion for 6 hours"
IF
patient is suggested Nesiritide

8.7.4
Consider medication: **Glyceryl trinitrate**
IF
patient is suggested nitrates

8.7.5
"Target dose: 300-600 ug sublingual, may be repeated if required OR 2.5 to 10 mg as sustained release tablets, two to three times daily"
IF
patient is suggested Glyceryl trinitrate

8.8 Anticoagulants
8.8.1
Consider medication: **Aspirin, Warfarin**
IF
patient is suggested anticoagulants

8.8.2

“Target dose: 50-325 mg/day”
IF
patient is suggested Aspirin

8.8.3
“Dose is highly dependent on patient status, usually 2-3 INR, once a day”
IF
patient is suggested Warfarin

8.9 Antiarrhythmics class III
8.9.1
“Target dose: 100-200 mg/day”
IF
patient is suggested Amiodarone

A9 Acute decompensation of congestive heart failure

9.1
Has sign or symptom of acute decompensation
IF
patient was previously diagnosed with CHF OR had signs and symptoms of CHF
AND
(patient has weakness OR patient has confusion OR patient has drowsiness
OR
patient has palor OR patient has cyanosis
OR
patient has cold and clammy skin
OR
patient has systolic blood pressure <90 mmHg OR drop of mean arterial pressure > 30 mmHg compared with usual values
OR
patient has narrow proportional pulse pressure ((SBP-DBP)/SBP < 25%))
OR
patient has filiform arterial pulse
OR
patient has oliguria
OR
patient has orthopnea
OR
patient recently had orthopnea
OR

patient has rales
OR
patient has jugular veins congested
OR
patient has hepatojugular reflux
OR
patient has ascites
OR
patient has peripheral edema
OR
patient has pleural effusion
OR
patient has hepatomegaly
OR
patient has tachycardia (heart rate > 100 beats/min) at rest
OR
patient has tachypnoe (>20 breaths/min) at rest
OR
patient has new-onset dyspnoea OR worsening of previously existed dyspnoea
OR
patient has new-onset cough OR worsening of previously existed cough

9.2 Acute decompensated congestive heart failure
9.2.1
Suggest diagnosis: **Acute decompensated congestive heart failure**
IF
Patient has sign or symptom of acute decompensation
AND
signs and symptoms are mild

9.3 Acute heart failure with hypertension/ hypertensive crisis
9.3.1
Suggest diagnosis: **Acute heart failure with hypertension / hypertensive crisis**
IF
Patient has sign or symptom of acute decompensation
AND
patient has high blood pressure
AND
patient has relatively preserved LVEF
AND
patient has pulmonary edema

9.4 Acute heart failure with pulmonary edema
9.4.1
Suggested diagnosis: **Acute heart failure with pulmonary edema**
IF
Patient has sign or symptom of acute decompensation
AND
patient has pulmonary edema
AND
patient has dyspnea
AND
patient has rales
AND
patient has orthopnea
AND
patient has respiration rate increased
AND
patient has reduced arterial oxygen saturation (<90%)
AND
patient does not fulfill the criteria of Acute heart failure with hypertension / hypertensive crisis

9.5 Low output syndrome / cardiogenic shock / severe cardiogenic shock
9.5.1
Suggested diagnosis: **Cardiogenic shock** OR **Severe cardiogenic shock**
IF
Patient has sign or symptom of acute decompensation
AND
patient has peripheral hypoperfusion
AND
heart rate > 60 beats/min
AND
(Systolic blood pressure < 90 mmHg
OR Drop of mean arterial pressure > 30 mmHg
OR low urine output (<0.5 ml/kg/h)
)

9.6 High output failure and septic shock
9.6.1
Suggest diagnosis: **High output failure**
IF

Patient has sign or symptom of acute decompensation
AND
patient does not have peripheral hypoperfusion
AND
patient has high heart rate
AND
patient has pulmonary congestion

9.6.2
Consider diagnosis: **Septic shock**
IF
Patient has sign or symptom of acute decompensation
AND
patient does not have peripheral hypoperfusion
AND
patient has high heart rate
AND
patient has Systolic blood pressure < 90mmHg

9.7 Right sided acute heart failure
9.7.1
Consider diagnosis: **Right sided acute heart failure**
IF
Patient has sign or symptom of acute decompensation
AND
patient has peripheral hypoperfusion
AND
(patient has Systolic blood pressure < 90 mmHg OR drop of mean arterial pressure > 30 mmHg compared with usual values)
AND
patient has increased jugular venous pressure
AND
patient has liver enlargement

A10 Heart failure cause and CAD risk factors

10.1 Heart failure cause
CAD
hypertension
valvular heart disease

myocarditis
cardiomyopathy
arrhythmia
pericardial effusion
diabetes mellitus
hyperthyroidism
hypothyroidism
anemia
amyloidosis
hemochromatosis
exposure to toxin
idiopathic (if nothing of those before is present)

10.1.1
“Consider CAD as one of two most probable causes of heart failure”
IF
patient has CAD
AND
patient has Heart failure

10.1.2
“Consider hypertension as one of two most probable causes of heart failure”
IF
patient has Heart failure
AND
patient has hypertension

10.1.3
“Consider valvular heart disease as a probable cause of heart failure”
IF
patient has Heart failure
AND
patient has valvular heart disease

10.1.4
“Consider myocarditis as a probable cause of heart failure”
IF
patient has Heart failure
AND
patient has myocarditis

10.1.5
“Consider cardiomyopathy as a probable cause of heart failure”
IF
patient has Heart failure
AND

patient has cardiomyopathy

10.1.6
“Consider arrhythmia as a probable cause of heart failure”
IF
patient has Heart failure
AND
patient has arrhythmia

10.1.7
“Consider pericardial effusion as a probable cause of heart failure”
IF
patient has Heart failure
AND
patient has pericardial effusion

10.1.8
“Consider diabetes mellitus as a probable cause of heart failure”
IF
patient has Heart failure
AND
patient has diabetes mellitus

10.1.9
“Consider hyperthyroidism as a possible cause of heart failure”
IF
patient has Heart failure
AND
patient has hyperthyroidism

10.1.10
“Consider hypothyroidism as a possible cause of heart failure”
IF
patient has Heart failure
AND
patient has hypothyroidism

10.1.11
“Consider arrhythmia as a probable cause of heart failure”

IF
patient has Heart failure
AND
patient has anemia

10.1.12

“Consider amyloidosis as a possible cause of heart failure”
IF
patient has Heart failure
AND
patient has amyloidosis

10.1.13
“Consider hemochromatosis as a possible cause of heart failure”
IF
patient has Heart failure
AND
patient has hemochromatosis

10.1.14
“Consider exposure to toxin as a possible cause of heart failure”
IF
patient has Heart failure
AND
patient has toxic exposure

10.1.15
“Cause of heart failure is idiopathic”
IF
patient has Heart failure
AND
patient does not have (CAD, hypertension, myocarditis, cardiomyopathy, arrhythmia, pericardial effusion, diabetes mellitus, hyperthyroidism, hypothyroidism, anemia, amyloidosis, hemochromatosis, esposure to toxin)

10.2 CAD risk factors
Hypertension
High LDL cholesterol (>100 mg/dl)
Low HDL cholesterol (<35 mg/dl male, < 45 mg/dl female)
High triglycerides (>200 mg/dl)
Diabetes mellitus
Family history of CAD
Smoking
Overweight (>25) / obesity (>30)

10.2.1
“Hypertension is a risk factor for CAD”
IF
patient has hypertension
AND
patient does not have CAD

10.2.2
"High LDL cholesterol is a risk factor for CAD"
IF
patient has LDL >100 mg/dl
AND
patient does not have CAD

10.2.3
"Low HDL cholesterol is a risk factor for CAD"
IF
patient does not have CAD
AND
(patient is male AND patient has HDL < 35 mg/dl OR patient is female AND patient has HDL < 45 mg/dl)

10.2.4
"High triglycerides is a risk factor for CAD"
IF
patient has triglycerides > 200 mg/dl
AND
patient does not have CAD

10.2.5
"Diabetes mellitus is a risk factor for CAD"
IF
patient has diabetes mellitus
AND
patient does not have CAD

10.2.6
"Family history of CAD is a risk factor for CAD"
IF
patient has family history of CAD
AND
patient does not have CAD

10.2.7
"Smoking is a risk factor for CAD"
IF
patient is smoking
AND
patient does not have CAD

10.2.8
"Overweight is a risk factor for CAD"

IF
patient is overweight
AND
patient does not have CAD

10.2.9
“Obesity is a risk factor for CAD”
IF
patient is obese
AND
patient does not have CAD

A11 HF diagnosis - expanded rule set

1.1
Patient has **heart failure signs**
IF ANY from:
systolic blood pressure (sitting) very low <85
systolic blood pressure (sitting) very high >140
diastolic blood pressure (sitting) very low <50
diastolic blood pressure (sitting) very high > 90
heart rate very low < 40
heart rate very high > 100
jugular veins congested
S3 sound heard
S4 sound heard
breath sounds absent
crepitations present
pleural effusion present
liver enlargement present
peripheral edema (sign) present

1.2
Patient has **heart failure symptoms** if
IF ANY from:
fatigue
dyspnea
orthopnea
paroxysmal nocturnal dyspnea
weight increase of more than 2 kg in the last month
peripheral edema (symptom)
fast heart beat (symptom)
irregular heart beat

A1
Patient has **final systolic diagnose**
IF
Patient has **diagnosed heart failure systolic positive**
OR
Patient has **diagnosed heart failure systolic negative**

A2
Patient has **final diastolic diagnose**
IF
Patient has **diagnosed heart failure diastolic positive**
OR
Patient has **diagnosed heart failure diastolic negative**

A3
Patient has **final diagnose**
IF
Patient has **final systolic diagnose**
OR
Patient has **final diastolic diagnose**

A4
Patient should **perform ecg test**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **final systolic diagnose**
AND
Patient has NOT **performed ecg test**

A5
Patient should **perform x-ray test**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **final systolic diagnose**
AND
Patient has NOT **performed x-ray test**

A6
Patient should **perform bnp test**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **final systolic diagnose**

AND
Patient has NOT **performed bnp test**

A7
Patient should **perform echo test**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT (**final systolic diagnose** AND **final diastolic diagnose**)
AND
Patient has NOT **performed echo test**

A8
Patient has **diagnosed heart failure systolic negative**
IF
Patient has **performed echo test**
AND
Patient has NOT (**left ventricular contractility decreased** OR **low LVEF**)

1.5
Patient has **diagnosed heart failure systolic negative**
IF
Patient has NOT **performed echo test**
AND
Patient has **performed ecg test**
AND
Patient has NOT (**anterior q waves** AND **left bundle branch block**)

A9
Patient has **diagnosed heart failure systolic negative**
IF
Patient has NOT **performed echo test**
AND
Patient has **performed x-ray test**
AND
Patient has NOT **cardiothoracic ratio high**

A10
Patient has **diagnosed heart failure systolic negative**
IF
Patient has NOT **performed echo test**
AND
Patient has **performed bnp test**
AND
Patient has NOT **bnp very high**

1.3
Patient has **diagnosed heart failure systolic positive**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has **performed echo test**
AND
Patient has (**left ventricular contractility decreased** OR **low LVEF**)

1.8
Patient has **diagnosed heart failure systolic positive**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **performed echo test**
AND
Patient has **performed ecg test**
AND
Patient has (**anterior q waves** AND **left bundle branch block**)
AND
Patient has **performed x-ray test**
AND
Patient has **cardiothoracic ratio high**
AND
Patient has **performed bnp test**
AND
Patient has **bnp very high**
AND
Patient has **ischemic heart disease**

1.4
Patient has **diagnosed heart failure diastolic positive**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has **performed echo test**
AND
Patient has **EA ratio low** OR **prolonged deceleration time**

A11
Patient has **diagnosed heart failure diastolic negative**
IF
Patient has **performed echo test**
AND
Patient has NOT (**EA ratio low** OR **prolonged deceleration time)**

A12
Patient has **heart failure unlikely**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **final diagnose**
AND
Patient has **performed ecg test**
AND
Patient has NOT **ecg abnormal**

1.6
Patient has **heart failure possible**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **final diagnose**
AND
Patient has **performed x-ray test**
AND
Patient has NOT **cardiothoracic ratio high**
AND
Patient has NOT (**pulmonary venous congestion** OR **pleural effusion**)

1.7
Patient has **heart failure probable**
IF
Patient has **heart failure signs** OR **heart failure symptoms**
AND
Patient has NOT **final diagnose**
AND
Patient has **performed x-ray test**
AND
Patient has NOT **cardiothoracic ratio high**
AND
Patient has (**pulmonary venous congestion** OR **pleural effusion**)

1.9
Patient has **heart failure positive**
IF
Patient has **diagnosed heart failure systolic positive**
OR
Patient has **diagnosed heart failure diastolic positive**

A13

```
Patient has **heart failure negative**
    IF
    Patient has **diagnosed heart failure systolic negative**
    AND
    Patient has **diagnosed heart failure diastolic negative**
```