BEN-GURION UNIVERSITY OF THE NEGEV
FACULTY OF ENGINEERING SCIENCES
DEPARTMENT OF INFORMATION SYSTEMS ENGINEERING

An Evaluation of a methodology for Specification of Clinical Guidelines at Multiple Representation Levels

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE M.Sc DEGREE

By: Erez Shalom

March 2006

ID: 027325372; email: erezsh@bgu.ac.il; Tel: 052-2542417

BEN-GURION UNIVERSITY OF THE NEGEV
FACULTY OF ENGINEERING SCIENCES
DEPARTMENT OF INFORMATION SYSTEMS ENGINEERING

An Evaluation of a methodology for Specification of Clinical Guidelines at Multiple Representation Levels

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE M.Sc DEGREE

By: Erez Shalom[1]

Supervised by: Prof. Yuval Shahar
And Dr. Meirav Taieb-Maimon

March 2006

[1] ID: 027325372; email: erezsh@bgu.ac.il; Tel: 052-2542417

Abstract

Background: We have previously developed a web-based Digital Electronic Guideline Library ("DeGeL"), which includes among others, the *URUZ* Markup tool, a web-based Knowledge Acquisition (KA) tool for structuring of clinical guidelines (GLs). In this research, a methodology for structuring of *procedural* and *declarative* knowledge of the GLs has been developed and evaluated in the context of the URUZ tool. This methodology includes specification and conversion of the GL's free-text representation by expert physicians (EPs), through an incremental representation into a machine comprehensible representation, thus enabling automated support to GL based care.

Objectives: To develop a methodology for acquisition of GLs by EPs, using tools for the structuring and specification in multiple representation levels, and to evaluate this specification process in qualitative and quantitative measures.

Methods: Three GLs, from three different clinical disciplines, were selected for use as the textual source for structuring by the EPs: *Pelvic inflammatory disease*, *Chronic Obstructive Pulmonary Disease*, and *hypothyroidism*. In the first stage, an Ontology Specific Consensus (OSC) was created. An OSC is a structured document that describes schematically the interpretation of the GL agreed upon by both the EPs and the knowledge engineer (KE), and includes the clinical directives of the GL and the semantic logic of the specification language. After learning the Asbru GL specification language and the DeGeL framework, and receiving some training in the URUZ KA tool, each of the EPs created a semi-formal Asbru markup using the URUZ, the OSC, and his own knowledge (for each GL, two markups were created by two different EPs). In order to evaluate each of the markups, a Gold Standard (GS) was created. The GS is also a semi-formal markup, which describes the best structuring of the GL, and is created by the EP and the KE together. Each of the markups created by the EPs was compared to a GS markup. In addition, subjective and objective measures were defined for qualitative and quantitative evaluation of the elicited knowledge in each of the markups: the subjective measures included several questionnaires to evaluate the EPs attitude regarding DeGeL framework, the specification language, and the usability of the URUZ tool. The objective measures were defined in two main categories: a *completeness* measure of the acquired knowledge, i.e., how much content from the GS exists (or not) in each of the semi-formal markups of each EP (for example, a predefined set of plans), and a *correctness* measure, i.e., how correct the acquired knowledge is from the aspects of 1) Clinical semantics and 2) Asbru semantics. The measures were used in some interesting incisions: for each knowledge role (KR) and group of KRs of the Asbru semantics, for each semi-formal markup, for each GL and for all GLs. The EP and KE collaborated to perform the evaluation process, using a graphical tool we had developed for this purpose, which enables online evaluation sessions by several users.

Results: There was no significant difference in the quality of the KRs acquired by the EPs**.** The completeness of the specification of all the KRs in all markups was high, with a mean of over 96% for all the GLs. In terms of correctness, there was a wide variability between the EPs (a mean weighted correctness of 0.6± 0.7 on a scale of [-1,1] with a range of [0.13,0.58]), but generally, the specifications quality was significantly higher than random. The variability between the EPs was noticeable in their clinical and semantics errors, whose type differed in each GL.

Conclusions: Given an OSC and a textual representation of a GL, any EP with some ontological training (not necessarily domain expert) can structure and specify its knowledge in a semi-formal representation completely. However, to structure and specify it correctly, we should select an EP with good computational skills. A future research study should include the use of graphical authoring tools. Given those initial insights, a prototype graphical interface has already been developed, and will be evaluated in the future**.**

❖ *This thesis is dedicated to my loved wife Maya*
who supported and encouraged me all the way..☺

Acknowledgments

I would like to thank those who guided, helped, encouraged and supported me during this research, especially my academic supervisor, Professor Yuval Shahar, who believed in me, and supported me all these years, always willing to help whenever necessary, and who motivated me to develop my skills and to fulfill my own potential and more. In addition, I would like to give special thanks to Professor Eitan Lunenfeld without whom I probably would not have finished my thesis, for giving me this rare opportunity to collaborate with him, and to use the resources in his medical division for this research. He was always more than ready to contribute his experience and time to this research, with patience and good will. I would like also to thank Dr. Meirav Taieb-Maymon who helped me through the statistical analysis of the results, and my research colleagues at the Medical Informatics Laboratory for their support during the research (listed alphabetically): David Boaz, Maya Galperin, Efrat German, Denis Klimov, Akiva Leibovitch, Lior Limonad, Robert Moskovitch, Alon Mayafit, Gil Tahan, and Ohad Young. The medical experts and clinicians to whom I am grateful for contributing their knowledge and time all these years are: Dr. Guy Bar, Dr. Lawrence Basso, Dr. Mary Goldstein, Prof. Yair Liel, Dr. Akiva Leibowitz, Dr. Tal Marom, Dr. Susana Martins, Dr. Laszlo Vazzar, and Dr. Avi Yarkoni. I thank the Department of Information Systems Engineering at Ben Gurion University for assisting me financially and the Faculty of Engineering at Ben Gurion University that gave me extensions, and let me complete my research. Last but not least, I wish to thank my supporting family: my parents Rita and Itizk Shalom and my sisters Inbal and Gali and her family. Finally, I would also like to thank the funding bodies for the overall support of research projects, in general and of my research project in particular, which was made possible by NIH award No. LM-06806.

.

Table of Content

Term Vocabulary

ASM – Asbru Semantic Measure
CIG - Computer-Interpretable Clinical Guidelines
CM - Clinical Measure
COPD - Chronic Obstructive Pulmonary Disease
DeGeL - Digital electronic Guideline Library
EP - Expert Physicians
GL - Guidelines
GLs -Guideline Source
GS - Gold Standard
HypoThyrd - Hypothyroidism
KA - Knowledge Acquisition
KB - Knowledge Base
KE - Knowledge Engineers
KR - Knowledge Role
MLM - Medical Logical Modules
MQS - Mean Quality Score
OSC - Ontology Specific Consensus
PBW - Plan Body Wizard
PID - Pelvic Inflammatory Disease

Key Words

1. Clinical Guidelines
2. Decision support
3. Digital libraries
4. Knowledge Acquisition
5. Knowledge representation
6. Ontology
7. Markup

List of Figures

List of Tables

1. Introduction: Specification of Clinical Guidelines at multiple representation levels

1.1. Clinical Guidelines and the problems involved in acquiring them

Clinical guidelines (GLs) have been shown to improve the quality of medical care, and are expected to assist in containment of its costs as well [Grimshaw and Russel 1993].
During the past 20 years, there have been several efforts to support complex GL-based care over time in an automated fashion (see section 2.1). This kind of automated support requires formal GL-modeling methods. Most of the methods use knowledge acquisition (KA) tools for eliciting the medical knowledge needed for the knowledge roles (KRs) of the GL specification *ontology* (key concepts, properties and relations) that each method assumes, in order to specify it in a formal format (see section 2.2 for detail description of all tools and methods). Using the terminology used in the recent Stepper tool [Svatek and Ruzicka 2003; Ruzicka and Svatek 2004], there are two main approaches to GL specification: *document-centric*, i.e., start from a free-text document and map it to a given GL ontology, and *model-centric*, i.e., model the GL de-novo using a predefined ontology and computational model, and refer to the source text only for documentation.
In most of those tools, however, there are still some unresolved issues: 1) The GL specification process should support and facilitate collaboration and inherent iteration between two different types of users – Expert Physicians (EPs) and Knowledge Engineers (KEs). 2) The specification process of the GL into a formal language is not sufficiently smooth and transparent. The core of the problem involved in the specification of a large mass of free-text GLs into a formal machine readable format is that EPs cannot (and need not) program in GL specification languages, while programmers and KEs do not understand the clinical semantics of the GL. The specification process should therefore be done gradually through several intermediate, semi-structured phases. 3) A framework that deals with GLs represented in multiple ontologies is required. 4) Some of the GL's knowledge is implicit and must become explicit during the specification process. 5) Not all the methods support the structuring process by markup, that is, structuring the GL text by labeling portions of text using semantic labels from chosen target GL specification language, sometimes even modifying the text. 6) There is a lack of an overall comprehensive methodology for the GL specification process. 7) There is lack of a complete evaluation methodology to qualify the results of the specification.
A hybrid representation model was therefore developed by us, which combines optimally the relative skills of the KE and the EP and solves most of the problems listed above.

1.2. The DeGeL Framework and the Hybrid Representation Model

An architecture and set of tools, called the Digital electronic Guideline Library (DeGeL) [Shahar and Young et al. 2004] was developed to support GL classification, semantic markup, context-sensitive search, browsing, run-time application, and retrospective quality assessment. DeGeL enables collaboration and inherent iteration between two different types of users – EPs and KEs - and facilitates the specification process of the GL into a formal language. In addition, The DeGeL library supports multiple GL ontologies, in each of which GLs can be represented in a hybrid format.
One of the DeGeL framework tools is the web-based URUZ markup tool. URUZ uses the infrastructure of DeGeL's hybrid guideline representation model and thus enables the EPs and the KEs on different sites to collaborate in the process of GL specification and to mark

up the GL in any representation level: semi-structured (typically performed by the EP), semi-formal (typically performed by the EP in collaboration with the KE), and formal (typically performed by the KE). This incremental process is shown in Figure 1: The EP indexes and marks up the GL (in this study, markup means structuring the GL text by labeling portions of text using semantic labels from chosen target GL specification language, sometimes even modifying the text), creating semi-structured representation, and in collaboration with a KE creating semi-formal GL representation. Then, KEs use an ontology-specific tool to add executable expressions in the formal syntax of the target ontology. Thus, each GL is represented in one or more representations levels: free-text, semi-structured text, semi-formal, and fully structured representation. All of these GL representation levels co-exist and are organized in the DeGeL library within a unified structure - the hybrid GL representation

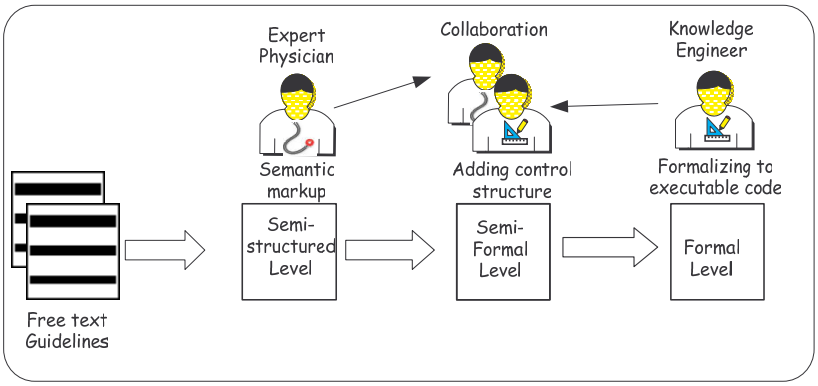


Figure 1. The incremental specification process in the URUZ markup tool**.**

In URUZ is embedded an Asbru-specific module, the plan-body wizard (PBW) which enables a user to decompose the actions embodied in the GL into atomic actions and other sub-GLs, and to define the control structure relating them (e.g., sequential actions). The PBW, used by EPs, significantly facilitates the final formal specification done by the KEs. DeGeL includes designed tools such as IndexiGuide for semantic classification of the GL, the Vaidurya context sensitive search and retrieval engine, and runtime tools such as the Spock guideline application engine (Shahar and Young et al. 2004). (See section 3.4 for all DeGeL tools.)

In the study, the *Asbru* language was used as the underlying guideline-representation language. The Asbru specification language includes semantic KRs organized in KR-Classes such as conditions (e.g., the filter condition, which represents obligatory eligibility criteria, the complete condition, which halts the guideline execution when some predefined criteria is true, and the abort condition, which aborts the guideline execution when some predefined criteria is true); control structures for the GL's body KR-Class (e.g., sequential, concurrent, and repeating combinations of actions or sub-guidelines), GL's goals KR-Class (e.g. process and outcome intentions), and the context KR-Class of the activities in the GL (e.g. actors, clinical-context).

As a web-based tool as part of DeGeL framework, using the infrastructure of the hybrid guideline representation model, URUZ enables EPs and KEs in different sites to collaborate in the process of GL specification and mark up the GL in any representation level: the textual representation level, the semi-formal level, and the formal level, and thus solves most of Problems 1 to 6 (see section 1.1). However, a methodology for the overall process of the specification and evaluation of the GLs still has to be defined.

1.3. A Methodology for Specification of Clinical Guidelines and its Evaluation

A methodology for knowledge acquisition (KA) by EPs, using tools for the structuring and specification of GLs in multiple representation levels and a methodology for evaluation of this specification process in qualitative and quantitative measures were developed.

The activities in the markup process include three main phases. 1) Preparations before the markup activities: choosing the specification language, learning the specification language, selecting a GL for specification, creating an Ontology-Specific Consensus (OSC), acquiring training in the markup tool, and making a Gold Standard (GS) markup. 2) During the markup activities: classifying the GL according to a set of semantic indices (e.g., diagnosis, treatment), and performing the specification process using the tools and consensus. 3) After markup activity: evaluation of the results of GL specification

For the evaluation of this methodology, three GLs, from three separate medical disciplines, were selected for use as the textual source for markups: Pelvic inflammatory disease (PID), Chronic Obstructive Pulmonary Disease (COPD), and Hypothyroidism. In the first stage, an OSC was created for each GL as an indispensable, crucial mandatory step. After learning the Asbru specification language and the DeGeL framework, and receiving training in the URUZ tool, each of the EPs created a semi-formal markup in URUZ using the OSC, and his/her own knowledge (for each GL two markups were created by separate EPs). In total 6 markups were created. In order to evaluate each of the markups created by the EPs, a GS was created. Each of the markups created by the EPs was compared to a GS (Figure 2). In addition, subjective and objective measures were defined for qualitative and quantitative evaluation of each plan, subplan and KR in each of the markups: the subjective measures included several questionnaires to evaluate the EP's attitude regarding the DeGeL framework, the specification language, the usability of the URUZ tool, and their subjective benefits. The objectives measures were defined in two main categories: a completeness measure of the acquired knowledge, i.e., how much content from the GS exists (or not) in each of the semi-formal markups of each EP (for example, a predefined set of plans), and a correctness measure, i.e., accuracy of the acquired knowledge according to the aspects of Clinical and Asbru semantics. Finally, the EP and KE collaborated to perform the evaluation, using a designated graphical tool developed for this purpose. This tool enables the measures of the markups from different sites and done by different users to be scored simultaneously.

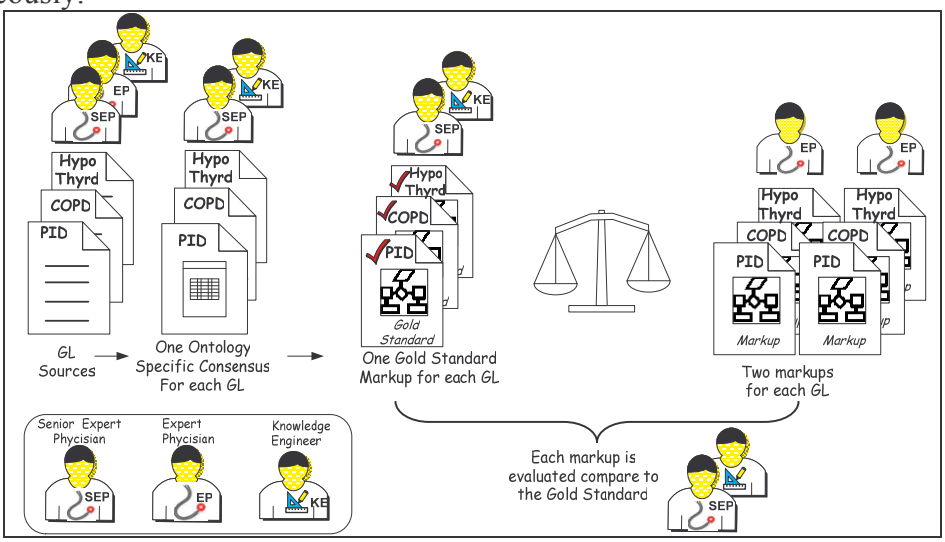


Figure 2. The Evaluation Design. Three GL sources were used. For each GL, an Ontology Specific Consensus and Gold Standard markup were created. For each GL two different markups made by separate EPs. Each markup was evaluated compared to the gold standard in clinical and semantic measures.

The evaluation of the methodology was very encouraging when using Asbru as the specification language, DeGeL as the representation framework, and URUZ as the markup tool. Table 1 summarizes the results for all GLs: In total, there were 196 different plans and subplans to be marked for the three GLs by the EPs: 106 plans in the PID GL, 59 in the COPD GL and 31 in the Hypothyroidism GL. In addition, a total number of 326 KRs were evaluated and scored in a clinical measure and an Asbru semantic measure for each plan and subplan in each markup for all GLs: 180 KRs in the case of the PID GL, 97 KRs for the COPD GL, and 49 for the Hypothyroidism GL. The completeness of the specification of all the KRs in all markups was very high, with a mean of 91% ± 0.11 for all the GLs. Regarding correctness, there was quite a variability between the GLs (mean of 0.6 ± 0.7 on a scale of [-1,1]), but the quality of specification for each markup was always considerably higher than zero. The levels of clinical and syntactical errors were different in each GL due to the differences in the level of detail in the OSC of each GL: as the OSC was more detailed in the clinical and Asbru aspects, the correctness measures were higher. The quality of complex procedural KRs (such as cyclical plan or specification of two plans in parallel or sequentially) acquired by the EPs was slightly lower than the declarative KRs (such as GL's filter condition) acquired.

As for the conclusions: it was found that the collaboration of an EP and a KE is crucial for successful formal specification of a GL. In particular, creating an OSC is an indispensable, crucial mandatory step before markup. Once an OSC and a textual representation of a GL are provided, any EP can in theory structure its knowledge in a semi-formal representation, but to specify it correctly, an EP with good computational skills should be selected. In addition, once the significance of the initial results of this research regarding usability and the difficulty of acquisition of procedural knowledge had been realized, a prototype graphical interface has been developed

1.4. The New Framework: The GESHER System

In parallel with receiving the preliminary results regarding the URUZ's low usability and cumbersome functionality, it was realized that a more intuitive framework should be developed which should be graphics-oriented and enable specification in multiple representation levels. Therefore, a new system for GL specification in multiple representation Levels (***GESHER***[1]) was designed and implemented. GESHER is a graphical client application, developed with Microsoft Dot.Net WinForm technology. GESHER enables access to centralized resources (such as the DeGeL library) as a client application, supports specification at multiple representation levels, enables collaboration between EPs and KEs, and is independent of any medical domain. In addition it enables the acquisition of both procedural and declarative knowledge, and handles multiple specification languages (GL ontologies). GESHER provides authentication and authorization services (directly or through the overall framework within which it is used), and accesses knowledge bases (KBs) in order to store and use the procedural and declarative knowledge of the GL. Finally, it uses standard medical vocabularies for referring to medical terms, to enhance the reusability of the GL's edited by the tool. An ongoing evaluation is currently being conducted with our collaborators with encouraging preliminary results

1.5. A Roadmap to this Thesis

We will now summaries the preceding sections and give an overview of the chapters to come.

- **Chapter 2**

In this chapter we will describe in detail the background of clinical guidelines: we will start with describing what clinical guidelines are and will explain the need for its automation and specification (Section 2.1) by developing computer-interpretable clinical guidelines (CIGs). Then we will describe and compare all main CIG methods and their KA tools (Section 2.2). In addition, we will describe and compare some attempts to evaluate KA tools, and will denote its drawbacks (Section 2.3). Finally, we will define all the unresolved issues related to guideline KA area. (Section 2.4)

- **Chapter 3**

After defining the problems relate to KA in chapter 2, in this additional background chapter we will present our solution for most of them, starting with our hybrid representation approach (Section 3.1), as part of the Digital electronic Guideline Library (DeGeL) architecture (Section 3.2). We will explain in detail as well the Asbru specification language used for this study (Section 3.3), and will describe the URUZ markup tool and all other DeGeL related tools (Section 3.4). Finally, we will explain the need for and overall methodology for specification and evaluation of clinical guidelines (Section 3.5)

- **Chapter 4**

In this chapter we will describe the methods we will use for this study which include to start with, a description of a methodology for evaluation of the specification process, its phases and its activities (Section 4.1). Then we will show the evaluation design of this study (Section 4.2). In addition, we will describe in detail the subjective measures (Section 4.3), the objective measures (Section 4.4) for this evaluation, and will present the Markup Evaluation Tool we used for facilitating this evaluation (Section 4.5). Finally, we will define the research questions and hypothesis of this research (Section 4.6).

- **Chapter 5**

After defining the research questions, we will show for each question its detailed result, its method of measurement, explanation and conclusion. The results in will be divided to three main groups: results for subjective measures (Section 5.1), results for objective completeness measures (Section 5.2), and results for objective correctness measures (Section 5.3).

- **Chapter 6**

As first result, we will present the new graphical framework for specification of clinical guidelines in multiple representation levels, namely *GESHER* . we will present its system and main philosophy(Section 6.1), its interface (section 6.2), its unique implementation of semi-formal representation view (Section 6.3), towards formal view (section 6.4).

- **Chapter 7**

Chapter 7 concludes the thesis with a summary of the main results and a discussion of possible directions for further research.

2. Background (I): Clinical Guidelines

2.1. Specification of Clinical Guidelines

Clinical guidelines (or Care Plans) are a powerful method for standardization and uniform improvement of the quality of the medical care [Grimshaw and Russel 1993]. According to the Institute of Medicine's (IOM) definition, clinical guidelines are "systematically developed statements to assist practitioner and patient decisions about appropriate health care for specific clinical circumstances" [Field, Lohr et al.1990] . Extensive evidence exists, conforming that state-of-the-art clinical guidelines (GLs) improves the quality of medical care, and sometimes even the survival chances of patients - a fact that had been rigorously demonstrated, while reducing its escalating costs [Micieli, Cavallini et al. 2002; Qualigini, Ciccarese et al. 2004].For our purposes as knowledge engineers (KEs), GLs are a set of plans, at varying levels of abstraction and detail, in some clinical domain (e.g., screening, diagnosis or management), and can be applied over limited or extended periods of time, on patients who have a particular clinical problem, need, or condition (e.g., fever of unknown origin; or therapy of insulin-dependent diabetes).

To implement guidelines within a computer-based clinical decision support system, GL representation is a critical issue [Wang, Peleg et al. 2002]. The recent review [De Clercq, Blom et al. 2004] has been identified main four areas involve in development GL-based decision support systems: (1) *Guideline modeling and representation* - which includes all building block represent the GL, the complexity of and level of abstraction of the GL, the knowledge types such as declarative knowledge that should be modeled separately, the GL documentation, the GL specification formal language and local adaptation methods for implantation in different institutes; (2) *Guideline knowledge acquisition* - which include the tools which facilitates the knowledge acquisition process by helping domain experts formulate and structure the knowledge used in GL, according to some specification language; (3) *Guideline verification and testing* – which include verification tests for detecting a various types of logical and procedural errors, and testing the GL in a simulation environments, in order to increase unambiguously, syntactically and semantically correctness of the GL; and (4) *Guideline execution* - which include GL interpretable by automatic parsers of GL execution engine, which should be application - independent in order to be used in multiple clinical domains and various modes of activation.

In this research, however, I will focus in particular on the aspects of GL representation, specification and acquisition. GL verification and execution are subject for other research.

2.1.1. The need for guideline specification

In order to utilize guideline knowledge in computerized systems, a representation model for the guideline knowledge must address issues such as:

- Include data structures for specifying the **procedural** knowledge of a guideline governing the application of its recommended actions (e.g., prescribe a drug, send a lab test). For example, synchronization constraints (e.g., do in parallel, in sequence) or explicit annotation of mandatory actions.
- Facilitate explicit representation of the clinical semantics of the guideline (e.g., target population, objectives). It is critical for the reasoning module to define all the knowledge it needs for its operation (e.g., to automatically select which patients are

eligible for the guideline) and leave nothing ambiguous. Typically, most of this knowledge of the **declarative** type.

- Support mechanisms for expressing various types of guidelines (e.g., alerts vs. clinical protocols) that differ considerably in complexity and comprehensiveness. For example, enable decomposition of a complex guideline into tree of less complex routines, each focused on a particular aspect of the guideline (e.g., diagnosis, follow-up).
- Include knowledge-roles (e.g., eligibility criteria, goals) that support the task for which the system is developed for (e.g., the task of runtime application versus the task of quality assessment).
- Offer customizable and reusable building blocks, common to clinical practice such as drug prescription or physical examination, which will ease the process of guideline authoring.
- Enable the interpretation of the content of the specified guideline by a computerized agent for purposes such as runtime application and other automated reasoning tasks (e.g., the task of quality assessment). Hence, the underlying representation model must be machine-comprehensible and must be accompanied by a suitable automated parser of some kind.

An implementation for automatic support of GL might be reminders and alerts, which can be viewed as "mini guidelines", and are useful mostly for representing a single rule that needs to be applied whenever the patient's record is accessed, as opposed to representation of a long-term plan [Peleg, Boxwala et al. 2001] . GLs and alerts are most useful at the point of care (typically, when the care provider has access to the patient's record), in particular when a specific care plan is prescribed by the care provider. Another aspects of major tasks involved in GL-based care that would benefit from automated support include knowledge-modeling tasks, such as specification and maintenance of GLs, runtime tasks such as application of GLs (which often need also patient data), and retrospective assessment of the quality of the application of the GLs. Other related tasks are search, retrieval, browsing, and visualization of relevant GLs, examination of the eligibility of one or more patients for a given GL, or the applicability of one or more GLs to a given patient. Thus, computer-based techniques could greatly assist in performing the tasks involved in guideline-based care.

Most clinical GLs, however, are text-based and not easily accessible to the care providers who need to match them to their patients and to apply them at the point of care. Similar considerations apply to the task of assessing retrospectively the quality of clinical-GL application. Some improvements have been made by publishing several of the GLs in an electronic format, such as HTML or PDF files. Yet, care providers, overloaded with information, rarely have the time, or the computational means, to assist them in utilizing the valuable knowledge, encoded in the GLs, during treatment. Therefore, there is a pressing need to facilitate automated GL specification, dissemination, application, and quality assessment. Another aspect is that machine-executable representations are crucial for providing computerized assistance, such as automated *execution* of the guideline, while for some other tasks, such as *search and retrieval* of relevant guidelines, text-based representations are of significant importance (and indeed, might be more useful).

All of these tasks cannot be supported without an appropriate infrastructure; since analyzing unstructured text-based GLs is not feasible, due to limitations of current technologies, such an infrastructure should include formal specification language (formal representation) of clinical GLs that can be parsed and executed by machines. We call such

representations machine comprehensible. (The term "comprehension" is used here in a strictly formal sense, not a cognitive one). Thus, the need to automate GLs implies, in practice, a need to convert the mass of free-text guidelines gradually into machine-comprehensible representations, in other words, the need to perform a GL structuring process. In addition, we require means to acquire the knowledge in order to support the gradual structuring process of the GL and apply it.

2.1.2. The task of Knowledge Acquisition

The process of acquiring the knowledge is called Knowledge Acquisition (KA).
Musen defined KA as "transformation of knowledge from the forms in which it is available in the world into forms that can be used by a knowledge system" [Musen 1993], and make a distinction between tacit knowledge (knowing how) and conscious, explicit knowledge (knowing that). When experts first solve a task, they have declarative knowledge about the application area of the task, but as they become experienced with the application the knowledge becomes procedural. Consequently, the tacit knowledge is the kind we would most likely want to incorporate into our knowledge base (KB). For example, in clinical GLs much of the knowledge comprising the GL is implicit. The goal of the KA process, performed by knowledge engineers (KEs) who are familiar with the KB domain, in collaboration with Experts physicians (EPs) who are familiar with the clinical domain and thus they are the domain experts, is to make the GL's knowledge explicit. This process, however, is extremely tedious, consumes considerable resources, and requires significant support.
EPs and KEs often suffer from a lack of communication. In the traditional view of knowledge-engineering, knowledge is supposed to flow from the EP, through the KE, to the KB [Feigenbaum 1984]. However, the different participants in the KA process rarely "speak the same language". Thus, the accuracy of the KB depends on the effectiveness of the KE as an intermediary. This problem has led to different approaches of interactions between the KE and the EP in the KA task. One of these approaches is to treat the KA task as model of problem solving and to use a conceptual model of the acquired knowledge that is based on an explicit model of a problem-solving method. Using these techniques requires that the KA program make assumptions on the basis of the knowledge a user has entered in order to operate it.
The first tool to use a conceptual model for expert-system knowledge, based on explicit models of problem solving, was ROGET [Bennett 1985], a program designed to assist KEs and EPs in developing an expert system. A dialog with its user, which was driven by terms and relations in a problem-solving model, was conducted, asking what kinds of evidence might be gathered during a consultation, and how that evidence might relate to possible hypotheses. Another approach to creating a problem-solving model with KA was to "hide" the model from the user. Using the tools, the knowledge was acquired directly from application specialists during the early stages of expert-system development, using an implicit model based on personal construct psychology. An example of such a system is ETS [Boose 1985]: After ETS conducted an initial dialog to elicit the constructs used by an application specialist to classify elements in the selected domain, the program would present visually as a repertory grid the knowledge that it acquired [Musen 1993].
All those approaches, however, required their users to define an application in terms of some predefined problem-solving method, and thus, programs which present the terms and relations of the predefined models of the general application tasks to be used, were developed in their place.
The Opal knowledge-acquisition tool [Musen, Fagan et al. 1987], a general model, was used by the ONOCIN [Shortliffe, Scott et al. 1981] system to acquire oncology protocols to

be used by the episodic skeletal-plan-refinement method. The specific plans were entered into OPAL, and then the basis of the treatment advice was offered by the ONCOCIN expert system. The limitation of the OPAL system was that it had only one domain-specific scope: cancer treatment. Thus, OPAL was later extended to the Protégé project [Shankar, Tu et al. 2002]. The Protégé project, headed by Mark Musen and his colleagues at Stanford University's Medical Informatics program, is a general framework for the design and automated generation of knowledge-acquisition tools, independent of task, domain, and problem-solving methods. Protégé supports reuse of domain knowledge by using domain ontologies to represent it. Ontology is "a reusable specification of conceptualization" [Gruber 1993]. By using domain ontologies, Protégé supports the configuration of a task model from a library of methods, and generation of task-specific KA Tools.

2.1.3. Automated Support of Guideline Based Care

During the past 15 years, there have been several efforts to support GLs over time in an automated fashion, and medical-knowledge representation standards were developed. Several approaches to the support of guideline-based care permit hypertext browsing of guidelines via the World Wide Web [W3C 2005], but do not directly use the patient's electronic medical record. Several simplified approaches to the task of supporting encoding GLs as elementary state-transition tables or as situation-action rules dependent on the electronic medical record, was attempted using the Arden syntax [Sherman, Hripcsak et al. 1995]. An established (ASTM) medical-knowledge representation standard, the Arden Syntax[Hripcsak, Ludemann et al. 1994], represents medical knowledge as independent units called Medical Logical Modules (MLMs), and separates the general medical logic (encoded in the Arden syntax) from the institution-specific component (encoded in the query language and terms of the local database).

Rule-based approaches, however, typically do not include an intuitive representation of the GL's clinical logic. In addition they have no semantics for the different types of clinical knowledge represented, have no ability to represent easily the GL and its components as a higher, meta-level problem-solving knowledge, and cannot represent intended ambiguity (e.g., when there are several options and several pro and con considerations, but no single action is, or should be, clearly prescribed) [Peleg, Boxwala et al.]. Furthermore, they do not support application of GLs over extended periods of time, as is necessary to support the care of chronic patients. On the other hand, as Peleg et al also point out, such approaches do have the advantage of simplicity when only a single alert or reminder is required, and the heavier machinery of higher-level languages is not required and might even be disruptive. Thus, they might be viewed as complementary to complex GL representations.

Attempts to automate the support for reminders and alerts using individual rules such as MLMs have been shown to be very effective. For example, computerized reminders and alerts increase the use of preventive care in the outpatient setting [Fedson 1994]. Furthermore, research conducted by [Dexter, Perkins et al. 2001] showed significant improvements when computerized reminders were applied in the inpatients setting as well. Thus, we firmly believe that similar results can be achieved for more complicated and comprehensive GLs as well.

2.2. Formal representation of clinical guidelines

During the past 20 years, there have been several efforts to support automation of complex GLs over time, in order to interpret them, thus, they are called computer-interpretable clinical guidelines (CIGs). This kind of automated support requires formal GL-modeling methods. Most of the methods use KA tools for eliciting the medical knowledge needed for the knowledge roles (KRs) of the GL specification ontology (key concepts, properties and relations) that each method assumes, in order to specify it in a formal format. CIGs are being developed to support decision-making during clinical encounters.

In this section I will describe the main CIG methods and their KA tools. However, a very comprehensive comparison between most of those CIGs and their representations can be found in [Peleg, Tu et al. 2002] and in [Wang, Peleg et al. 2002].

2.2.1. ONCOCIN

- Developed by:
  Stanford Medical School in the 1970s by Ted Shortliffe and his team
- Reference:
  [Tu, Kahn et al. 1989]
- Description
  The ONCOCIN project was a therapy-advice system designed for use by physicians in the treatment of cancer patients. An oncology protocol is modeled as a skeletal plan consisting of a hierarchy of actions and their sequence. To represent the skeletal plan and the refinement knowledge, the system must represent the hierarchy of plan components, the sequence of plan actions, and the heuristic knowledge, represented as rules and tables that map the patient's responses to past therapies to modifications of the standard treatment actions.

2.2.2. DILEMMA

- Developed by:
  Advanced Informatics in Medicine project program of the European Commission
- Reference:
  [Herbert, Gordon et al. 1995]
- Description:
  DILEMMA began as a project within the 1991-94 AIM program of the European Commission to develop computerized decision support, particularly for prescribing drugs. DILEMMA is an object model that contains an activity hierarchy, with the state transitions specified for these activities. A protocol typically consists of actions and activities that need to be carried out in order to perform the appropriate clinical tasks. A protocol may also be a component of another protocol. When a protocol is implemented, a procedure, defined as a type of action, may be generated. This procedure assumes one of many action states (e.g., relevant, established, requested, accepted, cancelled). Subsets of transitions between states are predetermined (e.g., from requested to complete). A transition is further controlled by state-transition criteria. The state-transition criteria are responsible for controlling the sequences in which protocols are implemented. Criteria are evaluated to select the protocols that are relevant at any given time. The foundations of the DILEMMA project were laid by the LEMMA project, the main objective of which was to apply a "logic engineering" approach to cancer therapy. The logic-engineering approach was previously used to design and implement the Oxford System of Medicine, a decision support system for general practitioners.

2.2.3. EON

- Developed by:
  Stanford University Medical Informatics
- Reference:
  [Musen, Tu et al. 1996; Tu and Musen 1999]
- Description:
  EON is a component-based suite of models and software components for the creation of GL applications, in particular to support the eligibility-determination and runtime-application tasks. Developers using EON can use a general architecture to build systems that support automated reasoning about guideline-directed care. EON project funding ended in March 2003. The SAGE project is carrying forward some of the work [OpenClinical 2005].
- Main philosophy
  EON's GL model (called the Dharma model) defines GL knowledge structures such as eligibility criteria, abstraction definitions, GL algorithm, decision models, and recommended actions. The GL algorithm is represented as a set of scenarios, action steps, decisions, branches, and synchronization nodes connected by a "followed-by" relation. EON provides three criteria languages to allow usability and medical expressivity: 1) a simple object-oriented language that clinicians can use to encode the majority of decision criteria; 2) a temporal query and abstraction language; 3) first-order predicate logic. An implementation of clinical practice guidelines, the ATHENA Hypertension Guideline Decision-Support System, uses the technology developed in the EON project in the clinical environment of the VA Palo Alto Health Care System, Palo Alto, CA [Goldstein, Hoffman et al. 2001]. EON's application to the treatment of AIDS is called T-Helper [Musen, Carlson et al. 1992].
- Application authoring Tool:
  Encoding of EON guidelines is done in the Protégé [Shankar, Tu et al. 2002] (now Protégé-2000) (Figure 3) knowledge-engineering environment. Protégé-2000 is an open-source tool that assists users in the construction of large electronic knowledge bases. It has an intuitive user interface that enables developers to create and edit domain ontologies. Numerous plugs-in provide alternative visualization mechanisms, enable management of multiple ontologies, allow the use of interference engines and problem solvers with Protégé ontologies, and provide other functionality. The Protégé user community has more than 7000 members [Noy, Crubezy et al. 2003] (see the summery of the Protégé-2000 evaluation in section 2.3.3).

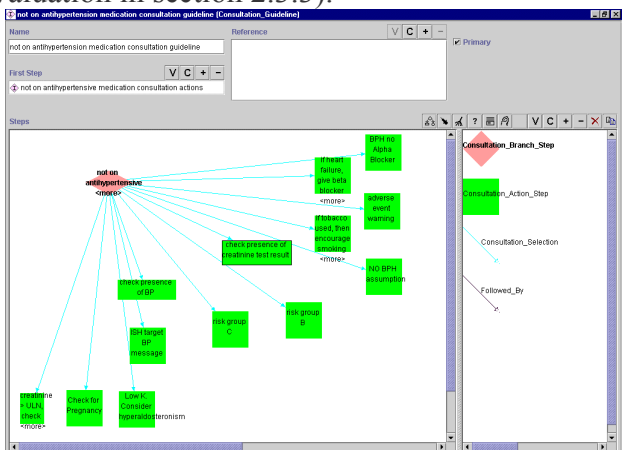


Figure 3. The interface of Protégé showing modeling in the EON framework**.** This figure showing modeling in the EON framework. This view is being used to author a guideline for managing hypertension. The guideline model being used in this application is Dharma, part of the EON framework [OpenClinical 2005].

2.2.4. Prodigy3

- Developed by:
  Sowerby Centre for Health informatics at Newcastle (SCHIN)
- Reference:
  [Johnson, .S.W. et al. 2000; Johnson, Tu et al. 2001; Peleg, Tu et al. 2002]
- Description:
  The Prodigy approach enables multiple entry points into a GL, using the concept of common clinical scenarios. PRODIGY I and PRODIGY II were implemented as extensions to proprietary UK electronic patient record systems. The PRODIGY system includes a GL model, which in PRODIGY II was used to implement guidelines for the management of acute diseases. The PRODIGY3 GL model uses a task-based formalism to represent the management of chronic diseases, particularly asthma and hypertension (Figure 4). Two vendors have integrated identical PRODIGY3 components into their clinical information systems for practitioners
- Main philosophy
  The model enables a guideline to be organized as a network of patient scenarios, management decisions, and action steps which produce further scenarios. Its purpose is to provide support for chronic disease management in primary care. It is built on the EON model and earlier versions of PRODIGY
- Application authoring Tool:
  Protégé-2000, see description in EON in section 2.2.3.

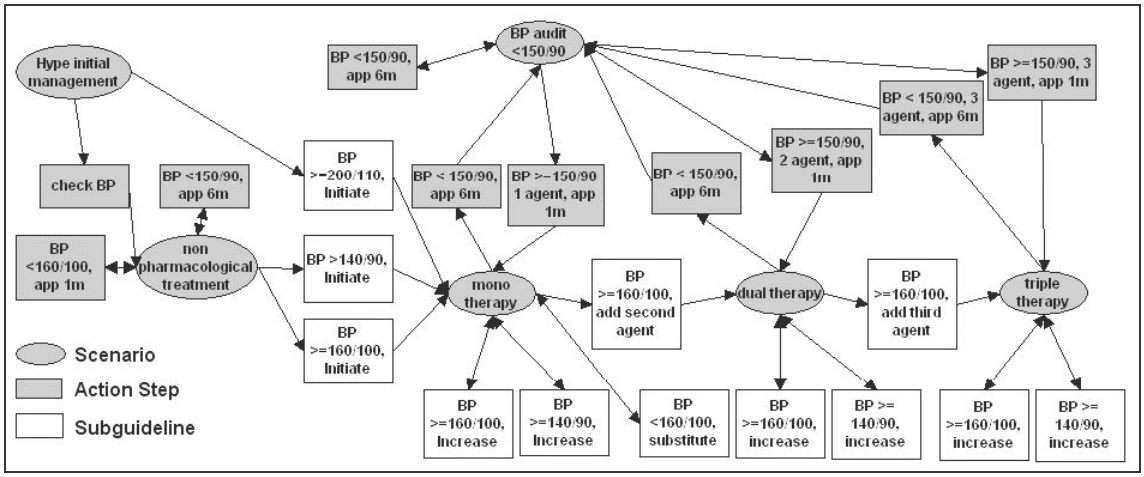


Figure 4. Modeling in the PRODIGY model. This view shows High level view of Hypertension guideline, with six top level scenarios [Johnson, .S.W. et al. 2000; Johnson]

2.2.5. Guideline Interchange Format (GLIF)

- Developed by:
  Groups at Columbia, Stanford, and Harvard Universities to support sharing of clinical guidelines among different medical centers
- Reference:
  [Ohno-Machado, Gennari et al. 1998; Peleg, Boxwala et al. 2000; Boxwala, Peleg et al. 2004]
- Description:
  GLIF version 3 (GLIF3) was designed to support computer-based execution by: 1) the inclusion of a superset of Arden Syntax's logic grammar as a formal expression language for specifying decision criteria and patient states; 2) a domain object model that enables GLIF3 steps to refer to patient data items, which are defined by a controlled terminology

that includes standard medical vocabularies (e.g., the UMLS) as well as standard models for the medical concepts (e.g., HL7 Reference Information Model - RIM) [Peleg, Boxwala et al. 2000; Peleg, Boxwala et al. 2001]; 3) using GELLO [Sordo, Ogunyemi et al. 2004], an object-oriented expression language to specify decision and eligibility criteria. GELLO was developed by InterMed in collaboration with the HL7 Clinical Decision Support Technical Committee, and is now considered an HL7 standard; and 4) using The Guideline Execution Engine (GLEE) [Wang, Peleg et al. 2004) to interpret guidelines encoded in the GLIF3 format and to integrate with clinical information systems for guideline implementation. (See the Summary of GLIF3 evaluation in 2.3.3)

- Main philosophy:
  GLIF3 represents guidelines in the form of a flowchart of guideline steps. Subclasses of guideline steps are action steps and decision steps, [Peleg, Boxwala et al. 2004] used to represent clinical actions and decisions, respectively. Decision steps contain several decision options. Patient state steps serve as entry points into the guideline, while also allowing for labeling of patient states.
  Branch steps and synchronization steps allow modeling of concurrent processes. In addition to the Guideline class, GLIF3 supports the use of the Macro class.
  GLIF3 was also used to evaluate the Guideline Execution by Semantic Decomposition of Representation (GESDOR) [Wang, Peleg et al. 2003] which was a model to share guidelines at the execution level such that guidelines can be shared even they are developed by different people at different locations and encoded in different formats.
- Application authoring Tool:
  Protégé -2000 (Figure 5), see description in EON in section 2.2.3.

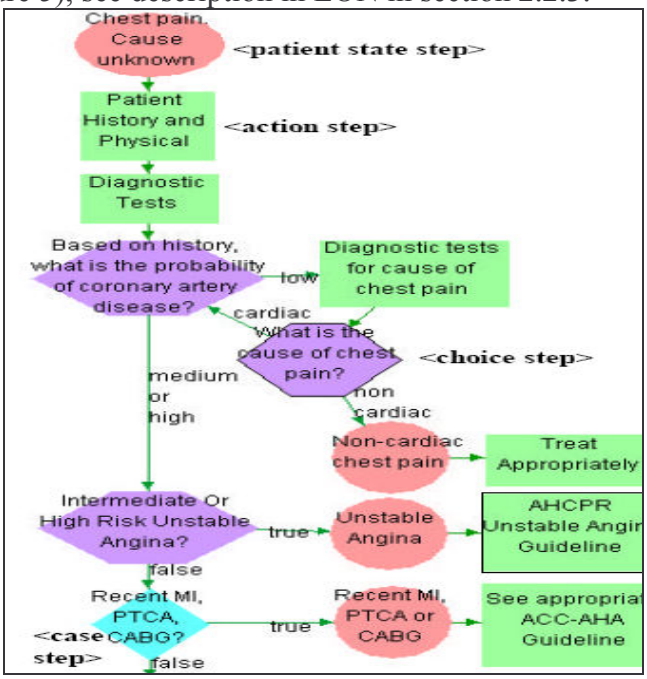


Figure 5. The interface of Protégé showing modeling in the GLIF format. This view shows part of a stable angina guideline written in the GLIF3 format and in the Protégé environment [Peleg, Boxwala et al. 2000]

2.2.6. Shareable Active Guideline Environment (SAGE)

- Developed by :
  IDX Systems Corporation, Apelon Inc., Intermountain Health Care, Mayo Clinic, Stanford Medical Informatics and University of Nebraska Medical Center.
- Reference:
  [Ram, Berg et al. 2004; Tu, Musen et al. 2004]

- Description:
  The SAGE project created infra-structure for implementing computable clinical practice guidelines in enterprise settings with the GL model. SAGE uses deployment-driven methodology for developing guideline knowledge bases which involves: 1) identification of usage scenarios of guideline-based care in clinical workflow; 2) distillation and disambiguation of guideline knowledge relevant to these usage scenarios; 3) formalization of data elements and vocabulary used in the guideline; and 4) encoding of usage scenarios and guideline GL using an executable guideline model.
- Main philosophy:
  SAGE recommendation-set formalism uses activity graphs, which are recommendation sets that allow specification of computational algorithms or medical care plans as processes consisting of: 1) contexts that are combinations of a clinical setting (e.g., an outpatient encounter in a general internal medicine clinic), care providers to whom the recommendation is directed, relevant patient states (e.g., patient age), and possibly a triggering event (e.g., patient checking into the clinic); 2) decision nodes that are loci of decision knowledge organized according to some decision model (e.g., a Boolean precondition for an action); 3) action nodes that encapsulate a set of work items that should be performed by either a computer system or a healthcare provider; and 4) routing nodes that are used purely for branching and synchronization of multiple concurrent processes.
- Application authoring Tool:
  Protégé 2000 (Figure 6), see description in EON in section 2.2.3.

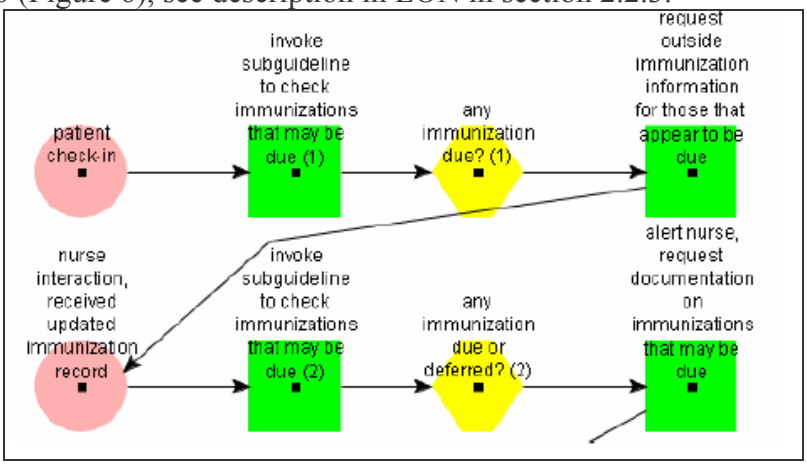


Figure 6. The interface of Protégé showing modeling in the SAGE format.. This view shows the top-level process specification in the SAGE immunization guideline. It defines how a guideline DSS should react to the events in the care process

2.2.7. PRO*forma*

- Developed by :
  Advanced Computation Laboratory of Cancer Research, UK.
- Reference:
  (Fox, Johns et al. 1998; Sutton and .J. 2003)
- Description:
  PRO*forma* combines logic programming and object-oriented modeling and is formally grounded in the R2L Language. One aim of the PRO*forma* project is to explore the expressiveness of a deliberately minimal set of modeling constructs. PRO*forma* software consists of a graphical editor to support the authoring process, and an engine to execute the guideline specification.

Main philosophy:

PRO*forma* supports four tasks: actions, compound plans, decisions, and enquiries. Plans are the basic building blocks of a guideline and may contain any number of tasks of any type, including other plans. Decisions are taken at points where options are presented, e.g., whether to treat a patient or carry out further investigations. Actions are typically clinical procedures (such as the administration of an injection) which need to be carried out. Enquiries are typically requests for further information or data, required before the guideline can proceed.

- Application authoring Tools:

1. **Arezzo**: clinical decision support software which enables the design, creation, and execution of clinical guidelines and patient care protocols that guide medical professionals with advice tailored for each patient individually (Figure 7). This is a commercial implementation of PRO*forma*. The Composer and Performer are products of InferMed Ltd. (http://www.infermed.com):

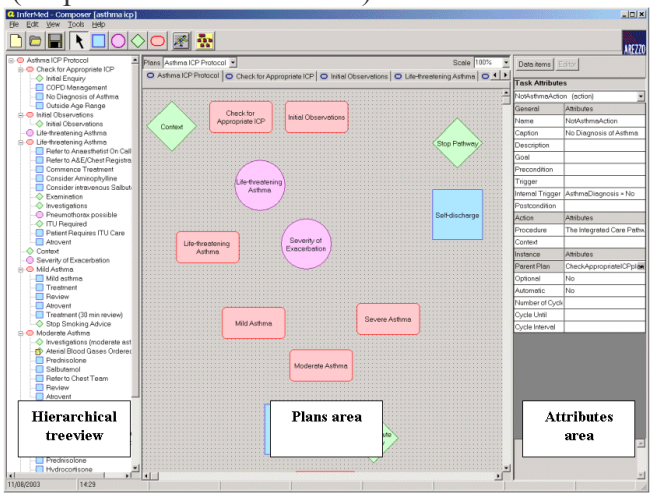


Figure 7. The AREZZO composer interface. AREZZO has a three-panel screen, with a hierarchical tree view of the guideline tasks in the left pane, task-authoring tool in the middle pane, and the attributes-authoring tool in the right pane

2. **Tallis:** Java implementation of PRO*forma*-based authoring and execution tools developed by Cancer Research UK(Figure 8). It uses the set of PRO*forma* task types (plans, actions, enquiries and decisions), and the detailed knowledge that is required to enact each component task which is entered via templates attached to each task type.

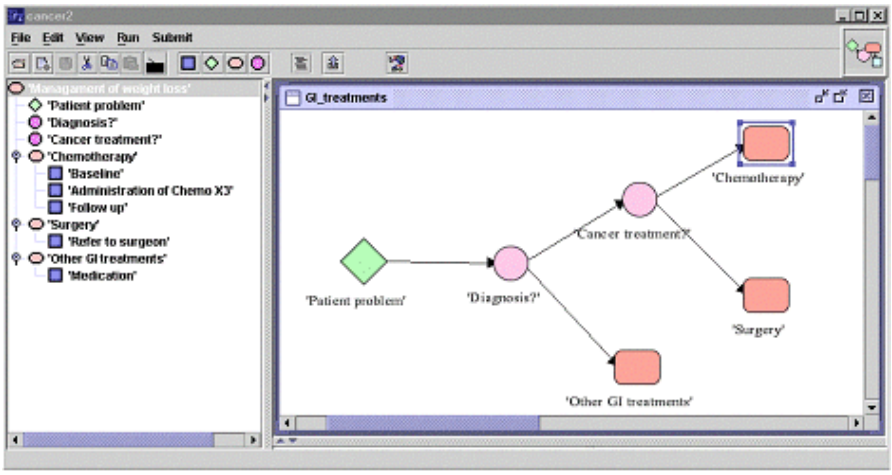


Figure 8. The TALLIS interface. This screen shows a network of tasks making up part of a (simplified) guideline for cancer diagnosis and treatment

2.2.8. GUIDE

- Developed by :
  Laboratory for Medical Informatics, Department of Computer and System Science, University of Pavia, Italy
- Reference:
  [Quaglini, Stefanelli et al. 2001; Ciccarese, Caffi et al. 2004]
- Description:
  GUIDE supports: 1) integration of modeled guidelines into organizational workflows; 2) use of decision analytical models such as decision trees and influence diagrams; 3) simulation of guideline implementation in terms of Petri nets. GUIDE considers issues such as patient data, the implementing facility's organizational structure, and resource allocation.
- Main philosophy:
  The recent architecture is the NewGuide, a GL management system for handling the whole life cycle of a computerized clinical practice guideline, which includes an editor to formalize GLs, a repository to store them, an inference engine to implement GLs instances in a multi-user environment, and a reporting system. NewGuide is based on two main levels: a central one (could be regional, national or international) and a local one (local health organization adopt one of the GLs to its unit).
- Application authoring Tool:
  "NewGuide" editor – a guideline authoring and execution environment (Figure 9). The main components of this editor are tasks (rectangles), and decision points (diamonds). Given the complexity of the care-delivery processes, we adopt a multi-level representation. Thus, some tasks in can be expanded into a lower, more detailed, level. The "NewGuide" editor produces four XML data structures containing the following information or knowledge: GEM for documentation, set of medical terms, set of abstractions and the GL flow

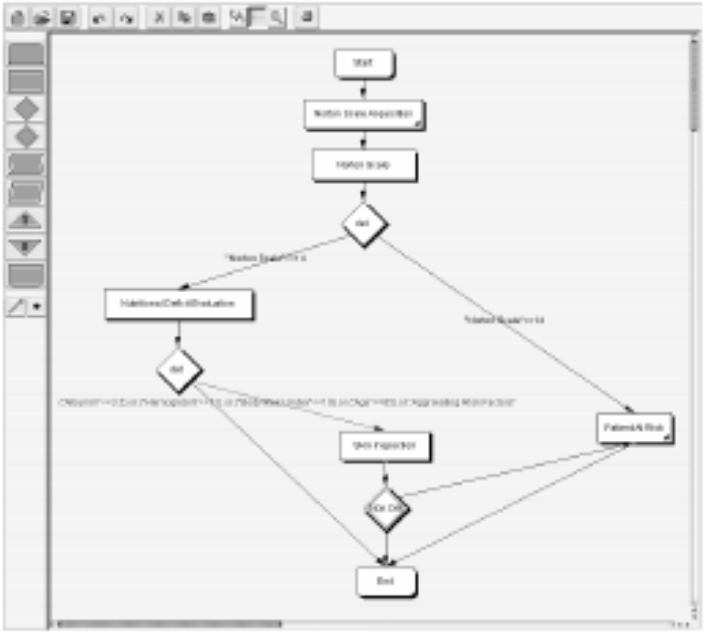


Figure 9. The interface of the "NewGuide" editor. This view shows the first level of pressure ulcer prevention GL in the "NewGuide" editor

2.2.9. PRESTIGE

- Developed by:
  The Prestige project
- Reference:
  [Gordon and Veloso 1999]
- Description:
  The Prestige Conceptual Guideline Model describes the kind of information, the concepts and the relationships between them, and provides an essential part of a meta-language for those responsible for writing the generic clinical scripts for particular scenarios, the protocol authors.
- Main philosophy:
  Prestige has a wide-ranging ability to represent knowledge about medicine, patients and careers, and the enterprises and personnel that care for them. It provides an essential part of a meta-language for those responsible for writing the generic clinical scripts for particular scenarios, the protocol authors. The Model has two major subdivisions: The first describes healthcare in general, and the second focuses on clinical protocols. The Model has been built and stored using a proprietary object-oriented CASE tool, SELECT Enterprise, which can produce skeleton software if required.
- Application authoring Tools:

1. **GAUDI** (Guideline Authoring and Dissemination Tool), which incorporates a terminology server and model (GRAIL, developed by GALEN) (Figure 10);
2. **GLEAM** (Guideline Editing and Authoring Module), which supports direct editing of an application knowledge base.

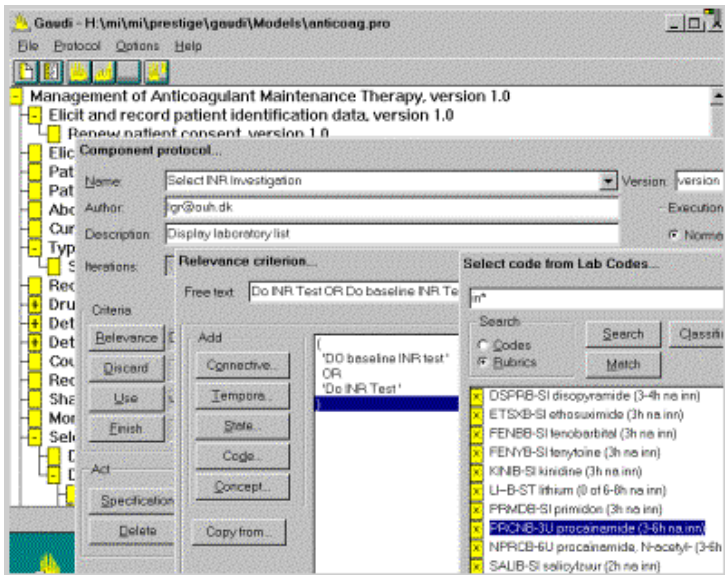


Figure 10. The GAUDI guideline authoring tool interface. GAUDI support the creation and editing of protocols that conform with the Generic Protocol Model.

2.2.10. GEM

- Developed by:
  Yale Center for Medical Informatics
- Reference:
  [Karras, SD et al. 2000; Shiffman, Karras et al. 2000]
- Description:
  Guideline Element Model (GEM) is a method that enables markup of the heterogeneous information contained in narrative clinical guidelines. Unlike other guideline representations, GEM encodes the strength of evidence for guideline recommendations.
- Main philosophy:
  The GEM hierarchy includes more than 100 elements. Major concepts relate to a guideline's identity, developer, purpose, intended audience, method of development, testing review plan, and knowledge components. GEM is intended to facilitate the translation of natural language guideline documents into a standard, computer interpretable format. GEM-encoding of guideline knowledge is pursued through a markup process that does not require programming knowledge. (See the summary of evaluation of this method in 2.3.3)
- Application authoring Tool:
  GEM cutter is an XML editor to facilitate markup of a guideline text document into GEM format. The main window (Figure 11) consists of three vertical panels. When a guideline document is imported into the application, it appears as a scrolling text document in the leftmost panel. In the middle panel, an expandable tree-view of the GEM hierarchy appears. A user classifies guideline contents by selecting text in the leftmost panel and clicking the insert button to place this text in the appropriate position in the GEM hierarchy displayed in the middle panel. GEM Cutter's rightmost panel provides definitions of elements and allows for text editing. Markup with GEM Cutter produces an XML file, the contents of which can be reused in a variety of ways.

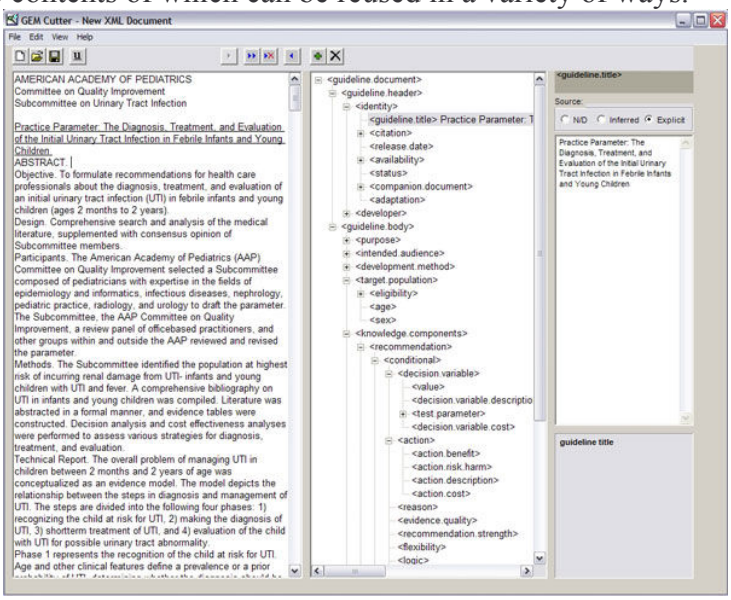


Figure 11. The GEM Cutter interface. The GEM Cutter interface. GEM Cutter is an XML editor that facilitates the transformation of guideline information into GEM format. Guideline source text is in the left panel, GEM ontology and its knowledge roles are shown in the middle panel. The portion of the text which associated with the selected knowledge role is shown in the right panel

2.2.11. GLARE

- Developed by:
  Dipartimento di Informatica, Università del Piemonte Orientale "Amedeo Avogadro", Alessandria, Italy, in co-operation with the Laboratorio di Informatica Clinica, Azienda Ospedaliera S. Giovanni Battista, Torino, Italy.
- Reference:
  [Terenziani, Montani et al. 2002; Terenziani, Montani et al. 2004]
- Description:
  GLARE (GuideLine Acquisition, Representation and Execution) is a domain-independent system for the acquisition, representation, and execution of clinical guidelines. GLARE is based on the assumption that knowledge in the clinical guidelines is independent of its use (e.g., support, evaluation, etc.), so that it is convenient (at least from the knowledge engineering point of view) to distinguish between the problem of acquiring and representing clinical guidelines and the problem of "using" them (e.g., "executing" acquired guidelines on specific patients).
- Main philosophy:
  GLARE distinguishes between atomic and composite actions. There are four different types of atomic actions: work actions (atomic actions), query actions (requests of information), decisions (embodying the criteria which can be used to select among alternative paths in a guideline), and conclusions (the explicit output of a decision process). Composite actions are defined in terms of their components, via the has-part relation, and have a set of control relations that establish which actions might be executed next and in what order (sequence, controlled, alternative and repetition).
- Application authoring Tool:
  GLARE knowledge acquisition tool (Figure 12), which enables graphical elicitation and browsing of the guideline, and automatic consistency checking of temporal constraints. The following figure shows part of the guideline for gallbladder stones treatment. The left part of the figure displays the window representing the general structure of the guideline. Each node represents an action, and each action has as sons the subactions composing it. The right part of the figure shows the window used to acquire the control relations between the components of composite actions. Each subaction is represented as a node in the graph.

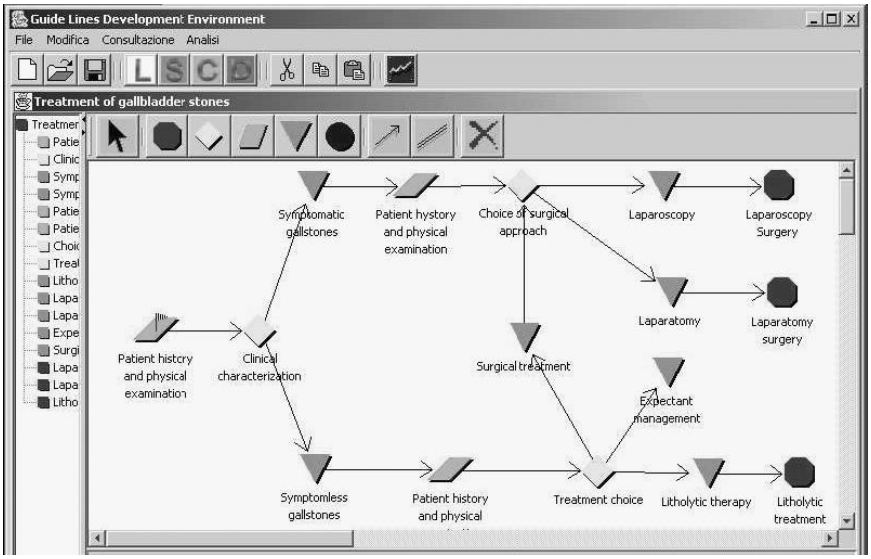


Figure 12. The graphical interface of the GLARE model. This view shows part of the gallbladder stones treatment guideline, represented through the GLARE acquisition module graphical interface

2.2.12. GASTON

- Developed by:
  the Department of medical Informatics of Maastricht University and the Signal Processing Systems group of the Eindhoven University of Technology.
- Reference:
  [De Clercq and Hasman 2004]
- Description:
  Gaston is a methodology and a framework that facilitates the development and implementation of computer-interpretable guidelines and guideline-based decision support systems. The overall goal of this approach is to improve the acceptance of computer-interpretable guidelines and decision support systems in daily care by facilitating all phases in the guideline development process.
- Main philosophy:
  The Gaston framework consists of: 1) a guideline representation formalism that uses the concepts of primitives, Problem-Solving Methods (PSMs) and ontologies to represent guidelines of various complexity and granularity and different application domains; 2) a guideline authoring environment that enables guideline authors to define guidelines; and 3) a guideline execution environment that translates defined guidelines into a more efficient representation, which can be read in and processed by an execution-time engine.
  The guideline representation formalism uses a frame-based model as an underlying mechanism. The formalism is non-monolithic, meaning that it can be extended with additional classes to capture new guideline characteristics.
- Application authoring Tool:
  Gaston authoring environment (Figure 13)

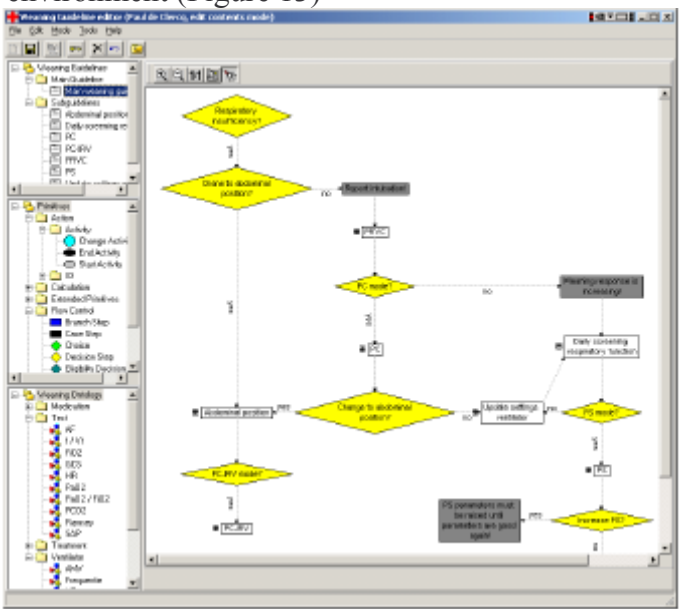


Figure 13. The interface of the Gaston KA-Tool. This view shows part of an ICU ventilation (weaning) guideline, entered in the Gaston KA-Tool The authoring environment relies heavily on drag-and-drop techniques to build guidelines. Steps, conditions and actions are not entered as text, but constructed by selecting and items from the various panes (e.g., terminology-panel) on the left.

2.2.13. HGML

- Developed by:
  Under development by the Clinical Informatics Research Group, University of Medicine and Dentistry of New Jersey; Department of Computer Science Rutgers University
- Reference:
  [Hagerty, Pickens et al. 2000]
- Description:
  HGML - Hypertext Guideline Markup Language- is an XML/XHTML specification for the identification of condition and recommendation elements within text guidelines.
- Main philosophy:
  HGML defines tags for conditional and associated recommendation elements which can be identified in a guideline text using a markup editor. The correlation of conditional variables, subject to their constraints, to clinical data allows delivery of recommendations linked to their original context.
- Application authoring Tool:
  HGML uses guideline authoring tool (Figure 14) to markup a unique color to display condition, recommendation, variable and relation tags, which all appear here in gray. In addition, condition tags are underlined so that the viewer can easily identify elements that are contained within them. In addition, an integrated graphical tool allows the user to clarify complicated logical statements by manipulating operators, conditions and recommendations in a spatial 2D representation. In addition, the tool can provide a "tree view" that illustrates the steps necessary to evaluate the conditions that lead to a particular recommendation

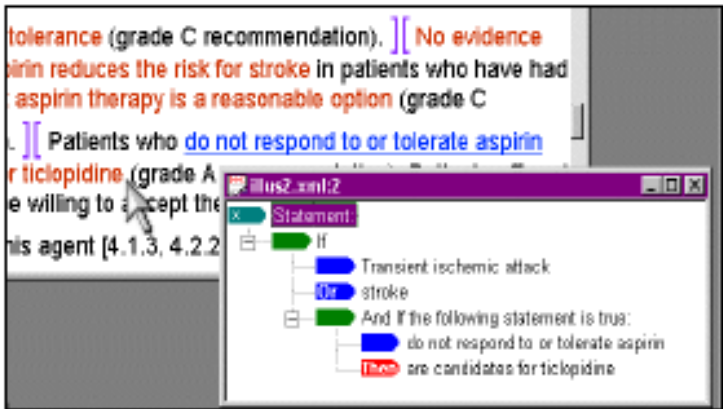


Figure 14. The HGML tagged elements. This view shows some illustrates of steps necessary to evaluate its condition. All elements in this window have been identified and tagged using the markup tool

2.2.14. Asbru

- Developed by:
  Asgaard project led by the Vienna University of Technology, Stanford Medical Informatics and Ben-Gurion medical informatics research center, Israel
- Reference:
  [Miksch, Shahar et al. 1997; Shahar, Miksch et al. 1998; Seyfang, Kosara et al. 2000]
- Description:
  In the **Asgaard** project [Shahar, Miksch et al. 1998], a comprehensive conceptual framework for clinical-guideline-based care was developed and an expressive guideline-specification formal language, called **Asbru**[2] whose focus is on representation of explicit intentions for the process and outcome, to better support the quality assessment tasks. Asbru can be used to design specific plans as well as support the performance of different reasoning and executing tasks.
- Main philosophy:
  During the design phase of plans, Asbru provides a powerful mechanism to express extended time oriented actions and plans caused by extended time oriented states of an observed agent (e.g., many actions and plans need to be executed in parallel or every particular time point). These plans are combined with intentions of the executing agent of plan. They are uniformly represented and organized in the guideline-specification library. During the execution phase an applicable plan is instantiated with distinctive arguments and state-transition criteria are added to execute and reason about different tasks. Asbru is unique in its ability to represent explicitly different aspects of the guideline, each of which is useful to one or more guideline-support tasks and the computational mechanisms that perform these tasks. The Asbru language represents a Clinical guideline with a clear and formalized semantic that enables the development of many computerized tools.
- Application authoring Tool:
  Three tools were developed by the Vienna University of Technology within the Asgaard project:
  1.AsbruView [Miksch, Kosara et al. 1998; Kosara and Miksch 2001]
  AsbruView is a tool to make Asbru accessible to physicians, and to give any user an overview over a plan hierarchy. AsbruView is based on visual metaphors to make the underlying concepts easier to grasp. This was done because not only is the notation foreign to physicians, but also the underlying concepts. This tool consists of two views: Topogical (TopView) and Temporal (TempView) views. TopView only shows the principal layout, and which parts of a plan are planned in sequence, in parallel, etc.

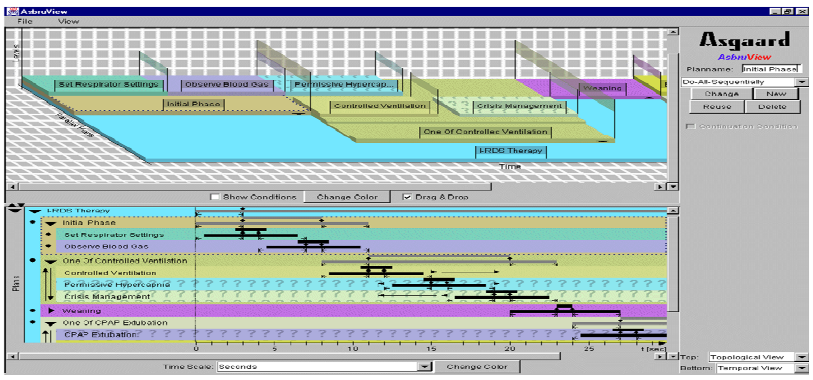


Figure 15. The interface of AsbruView program. This view shows the protocol for treating infant respiratory distress syndrome (I-RDS). The left/upper half panel shows the Topological View, the right/lower half panel shows the Temporal View.

[2] In Norse mythology, Asbru (or Bifrost) was the bridge from our world to Asgaard, the home of the gods

2.DELTA - Document Exploration and Linking Tool – (former The Asbru Guideline Markup Tool - GMT) [Votruba 2003; Votruba, Miksch et al. 2004]:

DELT/A (Figure 16) is the further development of the *Guideline Markup Tool (GMT)*. The aim of this project was to develop a tool that supports the transformation process of clinical guidelines from their original textual form (HTML) over an intermediate and a semi-formal representation (XML) to a formal representation that can be further used to verify the semantics of the guideline. Delta provides a relatively easy way to translate free text into Asbru. It does this by displaying both the original text and the translation, and showing the user which parts of the Asbru code correspond to which elements of the original text. This makes it easier not only to author plans, but also to understand the resulting Asbru constructs in terms of the original guideline. The GMT has two main features: links (two different files - either HTML or XML - side by side, and allows related parts in these two files to be connected using special links) and Macros **(** Macros combine multiple XML elements together with their attributes and can be used for simple construction of new XML documents). (For summary of evaluation of this method in section 2.3.3)

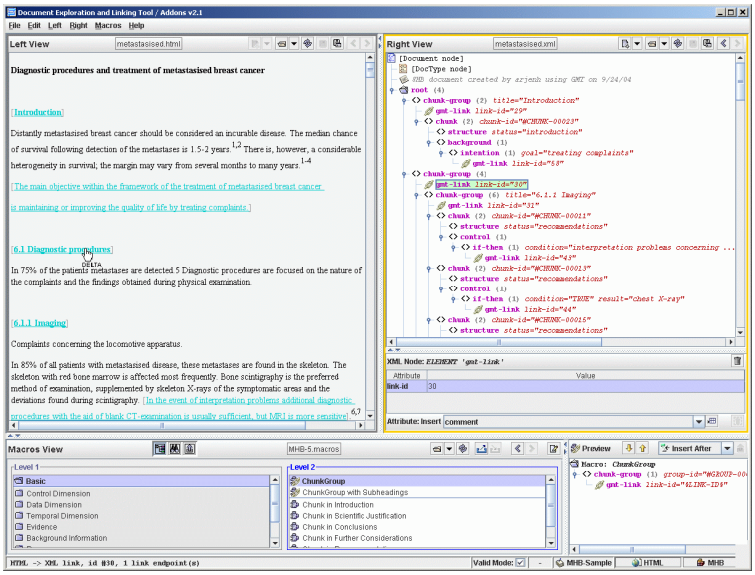


Figure 16. The interface of the DELTA tool. This view shows the Jaundice guideline. The original plain text of the Jaundice guideline is on the left, the corresponding Asbru tree on the right, and the Asbru structure aton the bottom. The marked parts of the original guideline and the XML representation are linked.

3.The Stepper Tool [Svatek and Ruzicka 2003; Ruzicka and Svatek 2004]

The *Stepper* tool (Figure 17) was developed to assist a knowledge engineer in developing a computable version of narrative guidelines. The system is document-centric: it formalizes the initial text in multiple user-definable steps corresponding to interactive XML transformations. The methodology in developing a guideline based on the idea that the system consists of multiple interconnected user environments: 1) for free-text markup (delimitation of initial 'knowledge' blocks); 2) for interactive step-by-step transformation of XML structure and content; and 3) for easy navigation along links (between source and target structures) across all transformation levels. Each instance of the model is actually a collection of components belonging to four categories: *procedural* component (roughly identified with the notion of scenario), *causality*, *goal statement,* and *concept definition*, since elements are gradually provided with tree structures of subelements; later, some of them (goals, scenarios) can also be e.g. aggregated

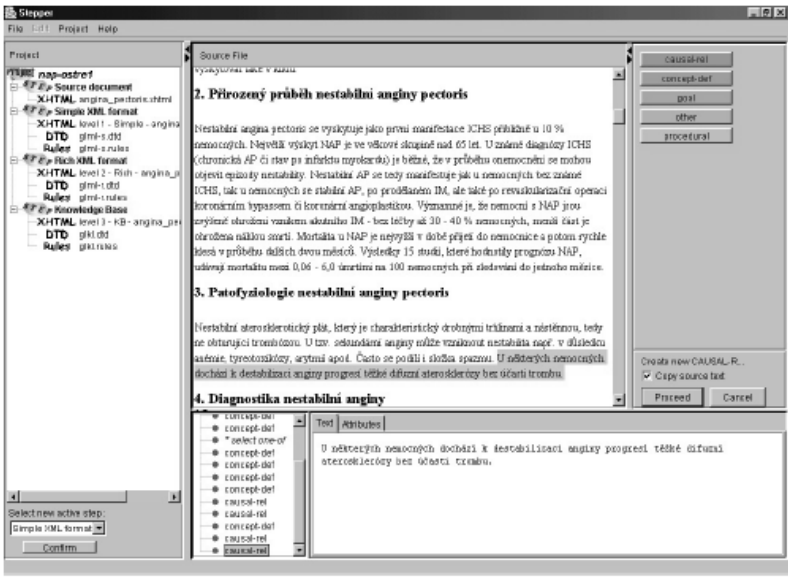


Figure 17.The interface of the STEPPER tool. This view shows extraction of basic components in step 1

4. **Automatic Extraction of Clinical Guideline Information** [Kaiser 2005]
   A recent interesting study suggests a multi-step approach using Information Extraction Transformation in which several heuristics extract knowledge from several guidelines from the specialty of otolaryngology. This transformation is multi-step and facilitates the formalization process using intermediate representations that are obtained by stepwise procedures. After defining guideline treatment processes (e.g., sequential, recurring), a heuristic for XHTML was developed to acquire them. The heuristic, using a pre-defined dictionary of terms, has for its first task detecting relevant sentences; then it has two subtasks: one detecting whether a sentence is the description of a process, an annotation, or whether it describes a negative action; and second, detecting relations between processes (a "process" is a sentence in guideline text). The underlying specification model it uses is ASBRU. Evaluation of this method was performed using 18 CPGs with 0.74 recall for task one and 0.94 for task 2 (recall measures the ratio of correct information extracted from the texts against all available information present in the text). The precision was 0.97 for task 1 and 0.84 for task 2 (precision measures the ratio of correct information that was extracted against all the information that was extracted.)

2.2.15. Comparison and Summary

Using the terminology of the recent Stepper tool [Ruzicka and Svatek 2004], there are two main approaches to GL specification: document-centric, i.e., start from a free-text document and map it to a given GL ontology, and model-centric, i.e., model the GL de-novo using a predefined ontology and computational model, and refer to the source text only for documentation.

In the model-centric approach, the relationship between the original document and the model is only indirect, and is mediated by the expert who is responsible for the initial step. This kind of model is "semantically closed" to the underlying model, and thus more appropriate for execution. A document-centric approach is more suited for coverage of information that does not fit well into the compact model (especially into the broader context of clinical care), and lowers the risk of leaving an important piece of information unspotted in the original text. Therefore, facilities are needed to translate the semi-structured representation of the guideline into a more formal executable representation. The document-centric tools are DELTA, STEPPER and GEM-Cutter, and HGML.

- **Document-Centric Methods**

  The DELTA tool is "specifically intended for knowledge engineers" [Votruba, Miksch et al. 2004] who have experiences with GLs and the specification language; the specification is done in one step, from text into formal representation, although it allows macros (an abstraction of common elements). The DELTA tool has an underlying language (ASBRU), but is oriented to a KE who is unfamiliar with the specification language, therefore can be used only for formal specification by KEs.

  The developers of the Stepper tool [Svatek and Ruzicka 2003] claims that the classification of high level knowledge portions has little to do with real medical experience because it is based on "common sense," so actually the KEs (who are not familiar with the clinical semantics) are responsible for the initial markup, whereas EPs are helping in the more advanced steps, such as extraction and refinement of the portions of the knowledge into concepts. The STEPPER tool has very detailed methodology regarding the markup process it is not formal specification-oriented in the sense that it does not have an underlying specification language. Although the STEPPER tool allows different parts of the GL to exist in different levels (not semantic levels of specification language), "by its nature" it does not support mixing of different levels of the same document. In addition, the first markup is performed by a KE, who is not familiar with clinical semantics.

  The GEM model has a very detailed declarative language which is good for documentation of the GL , but is not appropriate for execution: with its tool GEM-Cutter [Karras, SD et al. 2000], the EPs perform initial markups for elements which are associated to health services aspects (e.g., audience, support for clinical evidence), but, although this model includes some decision structures and links between elements, no collaboration with the KE is mentioned regarding further structuring with this tool, therefore, the GEM-Cutter too, does not include in its model the further refinement into more formal specification using KEs. However, in the their recent methodology [Shiffman, Michel et al. 2004] for making a requirement specification document, a KE should be involved when it comes to building a "executable statements", but again, the first stage of the markup is performed by EPs. It is not clear who performs the initial markup in the HGML tool; it is just noted as "authors" [Hagerty, Pickens et al. 2000].

  In addition, none of the document-centric tools is multi-ontology, and specific to one underlying language.

- **Model-Centric Methods:**

  Although GLIF3 (which was developed in Protégé-2000) is an effort to facilitate the specification process in multiple levels [Peleg, Boxwala et al. 2001] for EPs and KEs, most of the methods which use Protégé are addressed to a KE or an EP with some knowledge of the specification language (EON, Prodigy, SAGE), and therefore demand that the EP be familiar with the specification language, and to have high computentionla skills. In Gaston, GLARE, GUIDE and Pro*forma* a graphical interface was developed to acquire the procedural knowledge in a flowchart as the first step. In GLARE [Terenziani, Montani et al. 2002; Terenziani, Montani et al. 2004] and Proforma [Fox, Johns et al. 1998; Sutton and .J. 2003], it is the EP who acquires the knowledge, by using a set of icons for different "actions." In GASTON [De Clercq and Hasman 2004], however, a preliminary step, which is performed by a KE, is to define "domains and methods ontologies" using Protégé-2000, and store them in CLIPS format which is added to the KA tool as a plug-in. The "user" in GASTON (it is not clear whether this applies to the domain-expert or the KE) interacts with 3 types of interpreted managers: "Domain Manager, Guideline Manager, and Method Manager." After the KA process, its output "complied with primitives for execution".

In GUIDE [Quaglini, Stefanelli et al. 2001; Ciccarese, Caffi et al. 2004], the output of the specification task, which is performed by the "guideline developer" (again, it is not clear which expert performs the KA task), is translated into Petri-nets (using 9 relation tables) in "user transparent" fashion, and then automatically into the WPDL language, which is used as the underlying structure. In the NewGuide framework, the output of the KA process performed by an "editor" is XML which describes: 1) the GEM ontology; 2) medical terms; 3) abstractions; and 4) GL flow.

In all of these model-centric methods, however, there is no further intermediate process of refinement of the structure to be done by the KE, and thus the output of this process is the formal representation language in which it is executed. For procedural knowledge, this can be a very straightforward solution, but defining the declaratives or some more specific notion of the specification language, can constitute a not very simple task for the EP.

The only tools which are not specific to one specification language are the Protégé framework, in which one can specify multiple ontologies, and the GASTON KA tool which allows plug-in of pre-defined "domain ontologies".

- The following table and figure summarizes all approaches and the KA tool which they use:

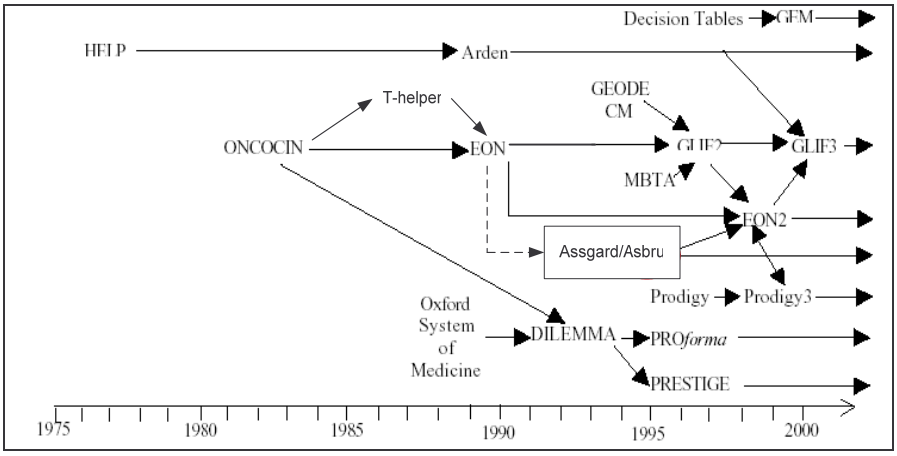


Figure 18. The history graph of guideline modeling methods. The guideline modeling methods are positioned on a time axis of the startdate of their development. Modified from [Elkin, Peleg et al. 2000]

Table 1 The Current knowledge acquisition tools for CIGs, their methods and approach. Note that Protégé-2000 is the main tool for most of the model-centric tools

Approach	Framework/ ontology	Knowledge Acquisition tool	WebSite
Model-centric	EON	Protégé 2000	http://smi-web.stanford.edu/projects/eon/
	SAGE	Protégé 2000	http://sageproject.net/
	Prodigy3	Protégé 2000	http://www.prodigy.nhs.uk
	GLIF3	Protégé 2000	http://www.GLIF.org
	PROforma	Arrezo,Talli	http://www.acl.icnet.uk/lab/proforma.html
	Prestige	GAUDI	http://www.ehto.org/ht_projects/initial_project_description/prestige.html
	GLARE	"CG AM" graphical interface	http://www.di.unipmn.it/
	GUIDE	NEWGUIDE	http://www.labmedinfo.org
	GASTON	Gaston authoring environment	http://www.medecs.nl/nl-NL/gaston.php
	Asbru	Asbru-View	http://www.asgaard.tuwien.ac.at/asbruview/index.html
Document-centric	Asbru	DELTA (GMT)	http://ieg.ifs.tuwien.ac.at/projects/delta/
	Asbru	Stepper	http://euromise.vse.cz/stepper-en/
	GEM	GEM-Cutter	http://gem.med.yale.edu/default.htm
	HGML	Guideline authoring tool	http://infolab.umdnj.edu/

2.3. Background: Evaluation of Guideline Knowledge Acquisition Tools:

2.3.1. Motivation for evaluation:

An evaluation can contribute greatly to understanding with what problems we have to cope when EPs try to elicit knowledge with some KA tool. In their book [Shortliffe, Charles et al. 2000] Shortliffe et al define the motivation for performing an evaluation of clinical information resources:

1. Promotional: Showing to the physicians that the resource is safe will benefit both patients and institutions and will encourage the use of this resource
2. Scholarly: In order to build the foundations of medical informatics as a discipline, a comprehensive evaluation should be required linking the structure, function, and effects of the information resources on clinical decision and actions.
3. Pragmatic: In order to know which techniques or methods are more effective or why certain approaches failed, we have to conduct an evaluation. In addition, we want to enable other researchers and developers to learn from previous mistakes.
4. Ethical: Before we use an information resource, we want to ensure that it safe and justify preferring it to other resources that compete for the same budget
5. Medico-legal: users need evaluation results to enable them to exercise their professional judgment before using systems.

As to our evaluation: the "information resource" is the knowledge acquisition tool that enables physicians to acquire knowledge of clinical guidelines according to some specification language (which in our case is ASBRU). Optimally, we would like to support several tasks related to the use of GLs, some of which are performed at the point of care: specification of the GL to start with, retrieval of the GL most applicable to the patient, correct application of the GL, and maybe even retrospective quality assurance. Such an evaluation would encourage the use of such tools for specification of the GL and demonstrate their benefits, and also increase the effect on the knowledge acquisition field, which is a sub-domain in the medical informatics discipline. We hope that this evaluation will be a benchmark for other ongoing evaluations in this field and hopefully increase the use of KA tools in the medical units for operational and educational purposes.

2.3.2. Evaluation of KA Tools - Related Problems:

"The main problem in evaluating KA techniques and tools is that they are designed to elicit quality knowledge from human experts. Therefore, when a tool is evaluated, the knowledge that the expert has delivered must be tested too..." [Shadbolt, O'Hara et al. 1999)

Conducting such a comprehensive evaluation is very complex and difficult. The statistical and measurement requirements are particularly hard to satisfy for the knowledge-acquisition systems, unlike general software products. According to Shortliffe et al (Shortliffe, Charles et al. 2000] there are five main problems involved in performing an evaluation of knowledge acquisition tools:

1. Amount of expertise – it is not easy to gather a large enough sample of experts together to give the results statistical significance.
2. Nature of acquired knowledge – it is difficult to compare, understand or validate the knowledge acquired during an experiment.
3. The acquired knowledge domain and tasks – it is necessary to discover the range of tasks and domains for which tools are useful, and to define over how many domains we need to run the evaluation, and how many and which tasks we need to run.
4. Scoping of the tool – it is difficult to isolate the value-added of the technique used in the tool from problems such as inadequate implementation, interface or platform

5. Quantification of the knowledge – it is difficult to select reliable and informative knowledge and to compare it in different ways.

To obtain a substantially large and diverse sample of subjects with a similar background requires significant amounts of resources. Obtaining the subjects for KA experiments is even harder when the tool is geared towards professionals in fields other than knowledge management, who may be less motivated to test the tools. The lack of a "gold standard" to compare the knowledge acquired with makes it almost impossible to measure the coverage of the knowledge and its value in terms of its inferential power. Another difficulty is to define how many and which domains and tasks should be used for the evaluation. If the tool is part of a complete framework, it is sometimes difficult to distinguish between problems which are intrinsic to the KA technique and problems caused by inadequate implementation, interface, or platform. In addition, whether a tool is to be defined in isolation or in a KA context with other tools, has to be defined. Another aspect is the nature of the evaluation results: should we conduct subjective or objective evaluation? Subjective evaluation involves questionnaires and interviews regarding the users' understanding of the nature of the domain, and usability testing. Objective evaluation usually involves qualitative and quantitative measurements regarding the completeness of the elicited knowledge, the clinical quality, and semantic quality of the acquired knowledge.

Fenton [Fenton 1991] classified three main classes of entity measurements in software engineering:

1. **Process measurement**: in the case of the URUZ markup tool, includes such parameters as usability, completeness, quality of clinical, and semantic knowledge;
2. **Resource measurement**: the subjects, personnel required for the study;
3. **Product measurement:** in the case of knowledge-based systems, the product is the result of the markups.

While processes and resources are quite intuitive to measure (by subjective questionnaires, for example), the product measurement is rather difficult because it measures the quality of the elicited knowledge. In such an evaluation, we should consider two main scales of measurements: the clinical scale, which measure the quality of the clinical content of the knowledge, and the specification language scale, which measure the quality of the semantics of the knowledge in the context of the specification language. Furthermore, we would like to measure in particular the specific type of clinical and semantic errors for these two scales.

2.3.3. Previous evaluations of KA tools summary:

In this section I will focus on the evaluation of four methods and their KA tools, two model-centric methods: Protégé2000 and GLIF3 (using Protégé 2000), and two document-centric methods: GEM and DELTA (former GMT) .For each method, I will focus on the following categories: amount of expertise, use of gold standard, the domain of the guideline, subjective results (for example usability), qualitative and quantitative results.

1.Protégé2000 [Noy, Crubezy et al. 2003]:

- Overview:

  This work evaluated the efficiency and quality of knowledge acquisition by domain experts, and how well the experts can retain their proficiency in using a knowledge-acquisition tool. Two groups, each consisting of two subjects, who were majors in the US Army (with no previous experience in KA) acquired knowledge about the domain constraints (typical organization of opposing-force) on military units according to task statements in the spirit of their daily activity.

- Evaluation Method:
  The subjects used different interfaces alternately: the regular Protégé-2000 interface[13], and HPKB tab which is a designated Protégé-2000 plug-in for this domain. Three knowledge acquisition sessions were performed: two sessions, one day after the other (in each session 90 minutes of instruction about the tool were given), and the third after two weeks. In the first session, one group (group A) elicited the knowledge with Protégé-2000 and the other group (group B) with the HPKB tab, in the second session vice versa. In the third session (after two weeks) the order was the same as in the first session. The manual on the organization of opposing-force units was used as the "gold standard" for determining the correctness.
- Evaluation criteria:
  1. Knowledge acquisition rate (num of KB changes per unit): Group A increased their rate and group B remained the same. After two weeks there was also no change
  2. Ability to fix internal errors in the KB: On the first day, with minimal experience using the tool and a low level of familiarity with the knowledge base and its organization, the subjects using the HPKB tab found 90% of the errors. The subjects using the generic version of Protégé-2000 were able to locate 81% of the errors.
  3. Quality of knowledge (how many errors appeared in the KB and how many changes in KB were correct/incorrect): The correctness rate ranged from 55% to 92% on the first day. However, on the second day and 2 weeks later, the correctness rate was uniformly above 97%.
  4. Error recovery rate (how many errors a subject noticed and corrected): On average, the subjects noticed and recovered from 32% of the errors they made. That is, the subjects later recovered from almost one third of the 7% of wrong steps that we mentioned earlier.
  5. Retention of skills after two weeks (to see if subject remembered how to use the tool): The subjects were able to find between 72% and 82% of errors that were introduced into the knowledge base. The data that they entered were more than 97% correct.
  6. Subjective opinion (usability questionnaires for each tool and for comparison between them): The results demonstrated that domain-specific user interfaces are useful, in particular at the earlier stages of using the tools.
- Results:
  The main result of the experiment was that domain experts, with limited computer experience, were able to use both Protégé-2000 and the enhanced domain-specific version to enter large amounts of complicated, highly interconnected data. In addition, they were easily able to find and correct errors in a very large knowledge base. They were able to do so after extremely short training sessions. Moreover, they were able to retain the skills they acquired during the first phase of the experiment, and to perform a similar set of tasks two weeks later, with little loss in productivity and no loss in the quality of the knowledge they entered. The effect of the enhanced user interface on the quality of knowledge entry was much less dramatic. Since Protégé-2000 itself provides a forms-driven, structured knowledge-acquisition facility, knowledge entered using Protégé-2000 is usually of very high quality. The forms "lead" users through the necessary steps, for example, making sure that they enter only the data of the allowed types, and giving immediate feedback on the data that the user just entered.
- Conclusions:
  If KA tools are intended for occasional short-term use, adding extra features that reflect the nature of the domain better would seem to be a worthwhile investment. Once the subjects became familiar with the knowledge base and the tool, regardless of the tool,

they were able to perform almost perfectly on a complicated task involving many types of highly interrelated information.

- Drawbacks:
  Drawbacks of this evaluation are that the study was limited by the resources that were available. Four subjects performed the experiments; thus the effect of individual differences with such a small number of subjects was a more likely source of errors, and it was decided to perform block-randomized tests.

2.GLIF3 (using Protégé 2000) [Patel, Branch et al. 2002] **:**

- Overview:
  This study was more in the context of knowledge representation, by evaluating the new version of GILF3 which was developed because of the limitations of the former language, GLIF2. In the evaluation performed by the interMed group, they were asked to generate GLIF2 and GLIF3 representations of the depression and thyroid guidelines in their natural, everyday work environment. Both subjects had extensive backgrounds in computer science: a medical informatician at the Harvard site who encoded the guidelines in GLIF2, and a medical informatician at the Stanford site who encoded the guidelines in GLIF3.
- Evaluation Method:
  The instructions given to the subjects were to use Protégé-2000 to encode the guidelines into the relevant version of the GLIF format, working from a graphic representation of the guideline and having access to the on-line text of the guideline. In addition, the subjects were instructed to verbalize their thoughts while they were encoding the guidelines.
- Evaluation criteria:
  What is the difference between the encoding of two clinical guidelines both into GLIF3 and into GLIF2 by two medical "informaticians"
- Results:
  Principal differences were identified in the encoding processes by the subjects who related to the formality associated with the encoding of guidelines in GLIF2 and GLIF3:
  1. The subjects were found to encode the guidelines using different representational constructs based on the way guidelines are modeled in GLIF2 and GLIF3.
  2. The representations of the guidelines encoded in GLIF3 were found to contain a greater level of detail than those found in the guidelines encoded in GLIF2. This is a result of the model used in GLIF3, which requires information to be encoded formally, thereby reducing ambiguity in the encoded guidelines
  3. The representation of temporal sequences and decision points in the guidelines also differed between the subjects, where the GLIF3 model avoids ambiguities found in the GLIF2 model.
  4. The analysis of the encoding of the guidelines revealed difficulties in the encoding process that may be related to differences between the GLIF2 and GLIF3 languages: in GLIF2 it was related to lack in formalism, while conversely, difficulties during the encoding in GLIF3 were mainly related to the formality of the language, which requires a greater level of detail in the encoding process.
- Conclusion:
  GLIF3's intended improvements in formality and expressiveness were achieved, including an improved ability for the accurate and efficient translation of clinical practice guidelines into a shareable electronic format.
- Drawbacks:
  No drawbacks were mentioned, but one drawback is probably the small sample of subjects.

3.GEM[Karras, SD et al. 2000**] :**

- Overview:
  This study evaluated the application of GEM by marking-up the guideline content using a hierarchical template that includes branches for identity, developer, purpose, intended audience, method of development, knowledge components, testing, and review.
  The subjects were a random sample of faculty, fellows, and residents, who represented informatics, pediatrics, and internal medicine at four institutions (Yale University, University of North Carolina, University of Alabama at Birmingham, and Johns Hopkins University). Subjects analyzed the American Academy of Pediatrics Practice Parameter on Neurodiagnostic Evaluation of the Child with a First Simple Febrile Seizure guideline.
- Evaluation Method:
  After an introduction to GEM of 15-20 minutes per individual, the participants were asked to copy and paste text from the guideline into pertinent elements in the template supplied as a Microsoft Word outline, and to add additional metadata as appropriate.
  Subjects were specifically instructed to analyze composite items into individual elements and to replicate branches of the template sufficiently to atomize guideline content. In addition, the subjects also completed a demographics and skills survey before the task, and a satisfaction survey afterwards.
  For each of the eight major branches of the GEM hierarchy, the number of element types used and the total number of elements were counted. Eight subjects marked-up the guideline, five of whom had no involvement in the development of the model. Three were members of the Informatics Faculty at three different institutions. Three of the subjects had pediatrics training, two were trained in internal medicine; the others were a neurologist, an anesthesiologist, and a graduate student. Satisfaction surveys were collected from the five subjects who were not involved in model development.
- Evaluation criteria:
  The main criteria included question regarding the percentage of unique elements used to represent content from the guideline, about the time taken by each of the subjects complete the task, and about the variation of markup.
- Results:
  The time to complete the task ranged from 90 to 169 minutes (median of 115). The total number of elements was highly correlated with the time to complete the task (Pearson's r of 0.823 with a significance of .012).
  The percentage of unique elements used to represent content from the guideline varied widely from user to user. There was disagreement as to the analysis of recommendations and their placement in the hierarchy. There was also disagreement among the participants as to whether a statement was conditional or imperative. This is surprising in that all subjects found the distinction between the two constructs to be "straightforward." There was a variation from one subject to another in how the same information was marked-up. Word count analysis noted a dichotomy among the participants. Some placed considerably more text into each element than did others who sought to abridge and abbreviate.
  Regarding the Survey, the subjects agreed that the GEM hierarchy was comprehensive enough to represent all (n=2) or most of (n=3) of the information content of the practice guideline. Most of the subjects found the "declarative" elements of GEM (such as Identity, Developer, Purpose, Intended Audience, Testing, and Revision Plan) "straightforward", but four of five found that analysis of "procedural" element (Knowledge Component elements) was confusing. Although most had no difficulty identifying actions and distinguishing conditionals from imperatives, three of the five participants reported difficulty with identification of decision variables. There was

considerable variation in the overall assessment of the straightforwardness of the task: three of the five found it to be straightforward, while the other two did not.

- Conclusion:
  Subjects from different backgrounds felt that the GEM model is sufficient to model the information content of the guidelines. However, there was substantial variation in their use of elements and the atomization of concepts, in particular, in analyzing and categorizing recommendations, and the content necessary for electronic guideline implementation.
- Drawbacks:
  The main drawback was small sample size of subjects, and the fact that only a single guideline was analyzed.

4.GMT (known as DELTA for Asbru ontology) [Votruba, Miksch et al. 2004**]:**

- Overview:
  In this evaluation, a small, qualitative study on the usability of the GMT Tool was conducted. The subjects were eight knowledge engineers, who were familiar with the Asbru language (two of them were even Asbru experts).
- Evaluation Method:
  The evaluation procedure consisted of three questionnaires.
- Evaluation criteria:
  1.A questionnaire assessing the skills of the participants.
  2.An exploration session, where the participants examined the functionality of the GMT.
  3.A questionnaire assessing the overall impression and the three views of GMT in particular.
- Results:
  The first phase showed that the sample was quite homogeneous, having similar skills (e.g., knowing various programming languages and software). However, the medical knowledge in the group, in particular knowledge about guideline-based care, was quite limited.
  The second and third phase confirmed that the three views (HTML, XML, Structure View) are very appropriate for authoring and structuring clinical guidelines, and for translating such clinical guidelines into a formal notation, such as Asbru. The linking features in both directions facilitated structuring guidelines' text, the retrieval of knowledge parts, and retracing of possible flaws and errors.
  Finally, the macros helped the participants to understand Asbru code and to write it better; however, this part needs more guiding features and design patterns so that Asbru guidelines can be written more easily.
- Conclusion:
  The participants rated the GMT as a very powerful and useful tool, which will support the implementation of clinical guidelines.
- Drawbacks:
  More functionality of the tool is needed

2.3.4. Summary:

Table 2 summarizes all the evaluation categories for the different methods and tools. It can be seen that there is no one approach that takes into account all the necessary stages for evaluating the markups: 1) making a consensus, 2) making a gold standard, 3) choosing the subjects and domain, 4) testing the subject's attitude regarding the specification language and its associated tools, 5) trying to find the concepts that most help the subjects in the markup and, in addition, 6) performing some usability testing of the tool. Most important is

a comprehensive plan for evaluating the markup outputs including detailed quantities measurement of 7) completeness and 8) correctness of the clinical and semantic aspects, all according to a gold standard.
In the **Methods** section (see 4) I will present my solution for all of these important considerations.

Table 2. Summary of the evaluation criteria for the different knowledge acquisition methods.

Method / Category	Protégé	GEM	GMT	GLIF3
Num of subjects	4 (2 groups of 2 subjects each)	8	8	2
Type of Subjects's expertise	Domain expert	Expert physicians	knowledge engineers	medical informaticians
Domain used	Constraints on military units	Pediatrics	None	depression and thyroid
Text Source	task statements	AAP guideline	None	depression and thyroid guidelines
Use of gold standard	Military manual	None	None	None
Subjective questionnaires results	None	None	Skills of the participants are quite homogenous	None
Usability measures results	Protégé has effective user interface	None	None	None
Qualitative measures results	Enhanced interface good for short-time learning	None	the three views mechanism of GMT very appropriate to KE authors	None
Quantitive measures results	* High ability to find errors *High KA rate when content in KB	*Time from 90 to 169 minutes *disagreement regarding the recommendations , their hierarchy ,the amount of text and "procedural elements" * agreement that GEM and "declarative elements are "straightforward"	None	*difference in representational constructs , representation of temporal sequences and decision points *difficulties in the encoding process
Completeness measures results	None	variation in percentage the amount of elicited unique elements	None	Ther is greater level of details in the results of GLIF3
Correctness Measure results	*correctnes of 55%-92% in first session, but above 97% in second and third session * 7% of subjects steps was wrong	None	None	None
Drawbacks	small number of subjects	small sample size, only a single guideline was analyzed	Lack of functionality	None

Note that there is no one approach that has considered all the different criteria as part of its evaluation design

.

2.4. The Unresolved Issues in Guideline knowledge Acquisition

Although there are so many tools from which to choose and work with, considering the comparison between them (see 1.2.15) there are still some unresolved issues related to KA task:

2.4.1. Physicians - Knowledge Engineer collaboration:

There is an unclear division of responsibility between the knowledge engineer (KE) and the Expert Physicians (EPs). The core of the problem is that EPs cannot (and need not) program in guideline-specification languages, while programmers and KEs do not understand the clinical semantics of the guidelines. Patel et al.[Patel, Allen et al. 1997] have shown that EPs interpret information differently from KEs and concluded that the developmental process for a GLIF-encoded representation must involve the active participation of both physicians and computer scientists at each stage in the evolution of the guideline's translation, although [Van Bemmel and Musen 1997] claims that this is the main bottleneck in the specification process. Thus, converting guidelines into machine-comprehensible formats must capitalize on the relative strengths of both experts, the clinical of the EP and the semantic of the KE, in order to support and facilitate collaboration amongst these two very different types of users, and the iteration inherent in such a process.
In most of the tools, however, there is no collaboration between the EP and KE, except in STEPPER, SAGE and GLIF3 where the specification is usually performed by domain expert who has some medical knowledge, but actually works as the KE

2.4.2. Incremental Specification:

The responsibility of the EP in the process of transforming the guideline is very heavy in that he should be familiar with the specification language. This process is usually done in one step, from the textual representation of the GL into a formal representation using some graphical user interface, and not in a gradual way. This kind of method in our experience is very difficult, not practical, and almost not possible for EPs who are not programmers. The specification process should therefore be done gradually through several intermediate, semi-structured phases, eventually arriving at a fully formal, machine comprehensible representation of the guideline.

2.4.3. Treatment of Multiple Ontologies:

Ontologies are a very central method in knowledge acquisition [Gruber 1993]. Usually the structuring methodology is focused on only one specific GL-specification language or ontology, and is not allowed specification according to multiple ontologies. It is therefore quite specific and has some limitations regarding the nature of the elicited knowledge.

2.4.4. Distributed Collaboration and Sharing:

Collaboration and sharing is essential for issues such as sharing the GL specification process among multiple developers or the subject authorization to access or modify the knowledge. However, most of the tools ignore these issues as they are standalone and therefore do not facilitate the need for sharing and collaboration of knowledge between different users and between different sites. None of the document-centric tools is web-based; they are all standalone.

2.4.5. Text Based Source:

Most of the electronic guidelines are currently represented in HTML format; therefore the ability to structure the guideline using ontologies labels, that is, perform semantic markup, is an important feature. Thus, we should be able to define for each segment of marked-up text, its semantic knowledge role. Naturally, only document-centric tools have this feature, namely starting from a text-based guideline and converting it into a formal structure, thus labeling the relevant text.

2.4.6. Knowledge Conversion:

Some of the guideline's knowledge is implicit in its nature, but clear only to the EPs authoring the guideline; this knowledge must become explicit during the conversion process. For example, procedural knowledge encoded in the guideline, such as clinical procedures, should be truly sharable and formal across multiple sites. Thus, the conversion process must support specification of terms from standardized medical vocabularies, which are well-understood everywhere, a task that the EP will perform best with practice.

In addition, [Coiera 2000] defined 4 group of knowledge conversion:

- Socialization (Tacit to Tacit) - "Sympathized knowledge": Share experiences to create tacit knowledge. Example: on-the-job training. Example: interacting with customers.
- Externalization (Tacit to Explicit) - "Conceptual knowledge": Articulate tacit knowledge explicitly: metaphors, concepts, hypotheses, models, writing.
- Combination (Explicit to Explicit) - "Systemic knowledge": Manipulating explicit knowledge by sorting, adding, combining, etc. Example: formal education.
- Internalization (Explicit to Tacit) - "Operational knowledge": Learning by doing, to develop shared mental models and technical know-how.

All of these groups of knowledge conversion should be expressed during the specification process. Generally, all tools are supporting this feature.

2.4.7. The overall process of guideline specification:

Once a textual guideline is selected, the EP or the KE, or typically both, create a formalized version of it in iterative fashion. This process can be very complex and raises multiple issues regarding its clinical and semantic aspects. Several conceptual questions can therefore be asked, such as: What are the steps for specification of a guideline using the KA tool? Are there any preliminary steps? Should we perform these steps using only the EP, or the EP together with a KE? How can we ensure the quality and completeness of the result? How do we decide what is the core knowledge we want to elicit? How does the underlying specification language change the process? How do we ensure minimal differences among different experts specifying the knowledge?

Therefore, a comprehensive, detailed methodology should be used taking into account all these considerations.

2.4.8. Lack of Complete Evaluation Methodology:

As opposed in chapter 2.3, there is no one, there is no one approach that takes into account all the necessary stages for evaluating the markups: first making a consensus, making a gold standard, choosing the subjects and domain, then testing the subject's attitude regarding the specification language and its associated tools, trying to find the concepts that most help the subjects in the markup, and performing some usability

tests of the tool. Finally, and most important a comprehensive plan for evaluating the markup outputs, including detailed quantities measurement of completeness and correctness of the clinical and semantic aspects, all according to a gold standard.

Summary:

Table 3 summaries all the mentioned problems in the different tools, and denote for each tool if there is implement for the problem in it. We can see that there is no one tools who answers all the problems. In particular, there is no tool which supports the incremental specification from text into machine comprehensible specification language the sharable (except the Stepper approach which not include a formal specification language as underlying language), enable sharing and collaboration between users as client application and has no comprehensive methodology for evaluation.
The following table (Table 3) summarizes all the problems of the different KA tools and methods:

Table 3. The problems of the different KA tools and methods, for each tool if it is denoted whether it answers the defied problem in the column header.

Approach	Method	Knowledge Acquisition Tool	EP-KE Collboration	Incremental Specification	Multiple Ontologies	Collaboration and Sharing as client web appliction	Text Based Source	Knowledge Conversion	Moethodology for Overall Process of Specification	Evaluation Methodology
Model-centric	EON	Protégé 2000			√			√		
	SAGE	Protégé 2000	√		√			√	√	
	Prodigy3	Protégé 2000			√			√		
	GLIF3	Protégé 2000	√		√			√		
	PRO*forma*	Arrezo,Talli						√		
	Prestige	GAUDI						√		
	GLARE	"CG_AM" graphical interface						√		
	GUIDE	NEWGUIDE						√		
	GASTON	Gaston authoring environment			√			√	√	
	Asbru	Asbru-View								
Document -centric	Asbru	DELTA (GMT)					√	√		
	Asbru	Stepper	√	√ -(1*)			√	√	√	
	GEM	GEM-Cutter					√	√		
	HGML	Guideline authoring tool					√	√		

* the Steppe tool does have a formal specification language as underlying language .Note that no tool is answering the need for an incremental specification (except Stepper) , the need for collaboration and sharing , and the need for an evaluation methodology.

Therefore, as I discuss in detail in the next chapter, we have developed a hybrid representation model, which combines optimally the relative skills of the KE and the EP and solves most of the mentioned problems.

3. Background (II): The DeGeL Library

To gradually convert GLs to machine-comprehensible representations, we have developed a *hybrid* (i.e., one that has multiple representation formats co-existing simultaneously), multifaceted representation, an accompanying distributed architecture, and a set of Web-based tools. The specification tools incrementally and in iterative fashion transform a set of clinical GLs through several intermediate, semi-structured phases, eventually arriving at a fully formal, machine comprehensible representation of the GL. The guiding principle to be described in this chapter is that EPs (if possible, throughout the world) should be transforming free-text guidelines into intermediate, semantically meaningful representations, while KEs should be converting these intermediate representations into a formal, executable representation.

3.1. The gradual conversion process

Underlying the various modules and tools i will be describing further on, is the guiding principle mentioned above: EPs use the tools to classify the guidelines along multiple semantic axes, and to semantically *markup* (i.e., label portions of the text by the semantic labels of the target ontology) existing text-based guidelines, thus creating a *semi-structured* format (which is still text-based). The EPs might even further structure the GL, possibly with a KE's assistance, into a *semi-formal* structure, which includes ontology-specific control-flow knowledge. *KEs* convert the marked-up text, or the semi-formal structure, into a *formal*, *fully structured*, machine-comprehensible representation of the target ontology, using an ontology-dedicated tool (Figure 19). All of the hybrid GL-representation formats co-exist and are organized in the DeGeL library within a unified structure, the *hybrid representation.* Part of the hybrid representation, shared by all hybrid guideline ontologies, is the *hybrid meta-ontology* [Shahar and Young et al 2004].

Note that different parts of the same GL might exist at different levels of specification (e.g., eligibility conditions might include also executable expressions, thus supporting automated eligibility determination, although the GL 's procedural aspect is still only semi-structured or in a semi-formal format). In addition, all specification levels are optional. Finally, if needed, new representation levels can be added.

Since EPs can rarely program (our experience over the past 4 years also indicates that they do not find control structures, such as sequential or parallel subtask execution, very intuitive), while KEs rarely understand all the hidden subtleties underlying the GL, it is necessary for both types of experts to interact at some point in the GL - specification process. This interaction usually happens when the EP (who is the domain expert) creates the semi-formal representation level, which includes specification of the control structures, assisted by the GL. Thus, the hybrid-specification process, which merges several grades of increasing formalization, intertwines the expertise of both types of experts to gracefully convert clinical guidelines into a machine-executable format. The conversion process is performed gradually using the following representation formats:

1. *Semi-structured Text* – snippets of text assigned to top-level target-ontology KRs, such as the eligibility criteria for applying the GL, or the GL's objectives. These roles would have different names in different guideline ontologies, of course.
2. *Semi-formal representation* – further specification of the structured text, adding more explicit procedural control structures, performed jointly by the KE and EP,

such as specification in explicit fashion of whether the actions are to be carried out sequentially or concurrently.

3. *Formal representation* – final specification performed by the KE, resulting with the GL converted to a machine-comprehensible format, executable by an appropriate runtime execution module specific to the chosen target guideline ontology. Thus, the output of our authoring tool(s) is a hybrid representation of a guideline which contains, for each guideline, or even for different sections (knowledge roles) within the same guideline, one or more of the above three formats.

These three current levels of hybrid structuring (or four, including the original free text) are in principle possible within all guideline-representation languages. For example, they were easily implemented within the context of the Asbru language, which happens to be the default guideline ontology in our architecture (see section 3.3).

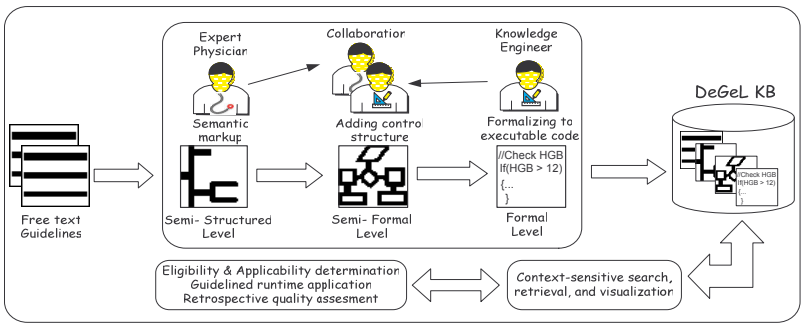


Figure 19. The incremental conversion process in the DeGeL architecture. Input free-text guidelines are loaded into a markup editor; expert physicians index and *markup* (structure) portions of the guidelines by semantic labels from a chosen target ontology, creating semi-structured and, in collaboration with a knowledge engineer, semi-formal guideline representations. Knowledge engineers use an ontology-specific tool to add executable expressions in the formal syntax of that ontology.

3.2. The DeGeL Architecture

We have developed a distributed, web-based architecture, the **Digital electronic Guideline Library** (**DeGeL**) [Shahar and O. Young et al 2004], which supports all of the design time and runtime tasks involved in guideline-based care. The DeGeL framework's guideline knowledge-base and various task-specific tools were designed to handle all of the hybrid guideline representation levels, enabling incremental specification in collaborative fashion between the EP and KE, and thus resolving problem 1 which was the need for infrastructure for collaboration between EPs and KEs and problem 2 which was the need for incremental process (see sections 2.41, 2.4.2).

The design for DeGeL architecture is not an arbitrary one. It incorporates insights from previous research projects, such as Asgaard [Shahar, Miksch et al. 1998]. There are three main components, in DeGeL's conceptual architecture (1) a permissions and authorizations manager component, responsible for generating user-profiles and controlling user access to DeGeL's guideline repository, (2) a guideline content manager, responsible for performing Create, Retrieve, Update and Delete (CRUD) operations on all knowledge entities (e.g., guidelines) stored in DeGeL's repository, and (3) a search & retrieval engine, responsible for performing text indexing and store semantic classification of guidelines as well as handling search queries processing. To support the specification of a guideline in one or more different GL specification languages, the DeGeL architecture includes a *hybrid* guideline *meta-ontology* (*meta* is used here in the sense of "*above*"). The meta-ontology is composed of two

components: (1)A *documentation ontology*, which specifies KRs common to all target GL ontologies, and defines the ontologies of the sources of the GLs and of the marked-up GLs, and (2) A *specification meta ontology* for describing a new target ontology, in order to enable adding it into the DeGeL (meta) knowledge base. Thus, DeGeL provide an XML schema that describes, for designers of existing or new GL ontologies, how to generate XML documents that conform to the DeGeL expected structure. These documents are instances of the specification meta ontology and describe particular target ontologies such as GLIF or Asbru .Implementing the hybrid-meta ontology in DeGeL enables uploading and structuring in more then one ontology, and thus resolving problem 3 which was the need for treating multiple ontologies (see section 2.4.3).

3.3. The Asbru Language and its Knowledge Roles

In the **Asgaard** project [Shahar, Miksch et al. 1998], there was designed an expressive guideline-representation language, **Asbru**. However, thorough this research some KRs were added to support additional requirements which were missing in the original Asbru. Following is the complete list Asbru guideline specification language, used for specification of the GLs:

- *Semantic indices* - The guideline classification according to the DeGeL's semantic axes (see 3.4.3)
- *Guideline knowledge* - Contains definitions of medical concepts which might be re used in several plans such as: clinical parameters, their definitions, and their classification criteria. (e.g., the definition of anemia in the context of a guideline for treatment of HIV)
- *Level of evidence* – The grades of evidenced based researches that a plan can be based on
- *Strength of recommendation* - The level we of recommendation of a plan
- *Actors* - Specifies who is responsible or taking part in performing the GL actions, for example: Nurse, gynecologist, etc.
- *Clinical context* - Specifies where in the clinical setting the patient is being seen outpatient clinic, for example: ER, ICU, extended care, etc.
- *Preferences* – supports the selection of a plan when more than one option exists. Preferences bias or constrain the selection of a plan to achieve a given goal. For example: (1) *select-method*, a matching heuristic to determine the applicability of the entire plan (e.g., *exact-fit* or *roughly-fit*); (2) *resources,* a specification of prohibited or mandatory resources relevant to the current plan (e.g., in certain cases of a pulmonary infection treatment, surgery is prohibited while antibiotics are mandatory).
- *Effects* – similar to the Preferences, it too support the selection of a plan by describing its expected implications if applied in a probabilistic manner and with temporal constraints. For example: (1) *argument-dependency*, describe the relationship between plan arguments and measurable effects by means of mathematical functions (e.g., in the context of a Gestational Diabetes Mellitus Type-II, GDM, the dose of insulin has a negative monotonic relationship to the blood-glucose level with likelihood of is 80 %); (2) *plan-effect*, describe

the overall effect of the plan on patient outcomes (e.g., in the context of a Gestational Diabetes Mellitus Type-II, GDM, a treatment plan might decrease the blood-glucose level with a likelihood of 0.97 after 10 to 60 minutes from the start of the plan).

- *Intentions* – supports intelligent retrospective quality assessment, by representing the guideline's intermediate and overall goals regarding care-provider actions and patient outcomes that should be maintained, achieved or avoided as temporal- patterns at multiple levels of abstraction. Four types of intentions can be specified: (1) *Intermediate state* - the patient state(s) that should be maintained, achieved or avoided during the application of the plan (e.g. weight gain levels should be slightly low to slightly high); (2) *Intermediate action* - the care-provider actions to perform during the application of the plan (e.g. monitor blood glucose once a day); (3) *Overall state pattern* - the overall pattern of patient states that should hold after the application of the plan (e.g. patient has an adequate glucose level); (4) *Overall action pattern* - the overall pattern of care-provider actions that should hold after the application of the plan (e.g. patient has visited dietician regularly for at least three months).

- *Conditions* – govern the lifecycle of the plan in perspective of its state transitions. Similar to intentions, conditions are also temporal-patterns at multiple levels of abstraction of patient state (e.g., if the patient has more then two episodes of severe anemia, each episode for a period of at least two weeks, since the start of the plan then abort the guideline) that should be evaluated during plan execution. Some of the condition should be evaluated during the selection phase of a plan:
    - *Filter-precondition* – must be true for the plan to transit from the *considered* state to the *possible* state (i.e., the eligibility criteria). If the filter-precondition evaluated to false, the plan transits to the *rejected* state and cannot be applied for that particular patient.
    - *Setup-precondition* – must be true before the plan transits to the *selected* state. Unlike the filter-precondition, if the setup-precondition is false when first checked it is possible to define a *waiting-period* in which it may turn true (e.g., the patient had a glucose-tolerance test taken no later then one month before starting the guideline). If the setup-precondition evaluates to false and the waiting period has elapsed, the plan transits to the *rejected* state. Thus, the setup-precondition is useful to ensure the existence of optimal circumstances prior to the activation of the guideline for a particular patient.

  Other conditions should be continuously monitored during plan execution:
    - *Suspend-condition* – if the suspend-condition holds the plan transits to the *suspended* state, hence its application is suspended and cannot continue for the meantime. Note that the application of the plan's subplans should be suspended as well. In addition, an optional recommendation, named *on-suspend*, can be used to specify an action to do when a plan suspends (e.g., in case of elevated TSH value suspend the plan and refer the patient to endocrinologist specialist).

- *Reactivate(Restart)-condition* – after a plan has been suspended, its reactivate-condition should be monitored to check whether the plan can be reactivated. If the reactivate-condition holds, the plan transits back to its previous activated state. Note that the reactivate-condition cannot be checked as long as the on-suspend recommendation has not finished yet. In addition, when a plan reactivates it should reactivate its subplans as well.
- *Complete-condition* – while the plan is in the *activated* state, the complete-condition should be monitored continuously. If the condition holds, the plan transits to the *completed* state and its application is terminated successfully.
- *Abort-condition* – while the plan is in either the *activated* state or *the suspended* state, the abort-condition should be monitored continuously. If the condition holds[3], the plan transits to the *aborted* state and its application has terminated unsuccessfully. In addition, an optional recommendation, named *on-abort*, can be used to specify an action to do when a plan aborts (e.g., admit patient to the ICU if patient's respiratory rate deteriorates).

- *Plan-body* – contains the recommended actions according to the guideline. As mentioned before, the Asbru language support the specification of complex controls structures using the plan-body knowledge-role. The types of plan-body elements include for example:
  - *Subplans* – allows to group two or more actions to be performed either in sequence, in a restricted order or in any order, or in parallel by setting the attribute *type* to one of the following values: *sequentially, any-order, parallel* and *unordered.* In addition, the subplans element contains a special element, also known as the *continuation-specification* (in Asbru 7.3 it was redefined as two elements: *wait-for* and *abort-if*), that enable to specify which of the actions are mandatory (i.e., required to complete successfully in order for the parent plan to complete successfully too).
  - *Plan-Activation* - activation of another plan
  - Simple Action - An atomic plan with simple semantics ,Suitable for defining plans with one action such as take drug
  - *Switch- Case* - When criteria has some possible values. For each value, a plan should be defined
  - *If-Then-Else* – branching statement to reflect various options based on either patient sate or plan execution.
  - *Cyclical-plan* – allows specifying an action that should be performed repeatedly. In order to control the start time, frequency settings and finish time of the repeated action the following elements are used: (1) *start-specification* - determines the time-point, based on patient state or plan execution state, at which the repeating action is executed for the first time; (2) *repeat-specification* - defines the times at which repeated actions take place (e.g., waiting periods between consequent

activations); (3) *finish-specification* – describes the time at which a cyclical plan should end (e.g., after fix number of activations).

- *To-Be-Defined* - The plan needed to be define later

There are other knowledge-roles in the Asbru ontology not mentioned in this review. For example, the Asbru ontology provides the knowledge-roles *library-definitions* and *domain-knowledge*.

In addition, each Asbru plan can store its own information, get information from other plans or pass information to other plans using mechanisms similar to ones used in modern programming language of variables and arguments of functions. The Asbru's constructs used for this purpose are:

- *Variables* - A variable is s value whose temporal dimension is ignored. Like variables in any programming language, it has scope constraints, local for variables declared for a specific plan and global variables for variables declared for a library of plans using the library-definitions knowledge-role.
- *Parameters* - As opposed to variables, parameters are observed over time. Their values may be based on other parameters values thus supporting abstracted parameters. Similar to variables, parameters also haves scope and can be declared or globally.
- *Arguments* – Are used to pass information to a plan or out of a plan. The Arguments of a plan have the same scope as local variables (i.e., they are valid for this plan only). There are two kinds of arguments, *input* and *output* arguments (i.e., similar to values passed and returned from functions).

Finally, the knowledge-role *Defaults* is used to store various definitions such as the typical duration of the time required to apply a plan, any temporal constraint on the application of the plan (e.g., when to start or finish the plan) and the optional recommendations to apply when the plan aborts or suspends. Note that these definitions can be override by a plan invoking another plan.

We have created a hybrid-Asbru ontology, whose semi-structured level is used by the EPs in the first phase of the conversion process. In the *Asbru semi-structured hybrid ontology*, we have included key entities such as *conditions*, *intentions, effects, preferences* and *plan body*, but left out low-level knowledge roles that require deeper understanding of Asbru semantics.
In the next chapter I introduce several Asbru–specific tools we have implemented for supporting conversion into the semi-formal and formal representation levels (see 3.4). We used the Asbru ontology in for demonstration of the current architecture's various aspects. It is the default ontology we used for the GL specification process. Note that the services supplied by the DeGeL framework are the same for all hybrid guideline ontologies with respect to the meta-ontology and the semi-structured text representation level. For example, we have also marked up guidelines using the GEM ontology, as well as by guideline ontology, specific to the needs of the CIC (*Clinical Information Center* in Tel Aviv, Israel) organization. Furthermore, the overall guideline-specification workflow is essentially independent of the particular target guideline ontology.

3.4. The URUZ markup Tool And Related DeGeL Tools

3.4.1. The URUZ Interface

The *Uruz* , a client web-based guideline markup tool (Figure 20) [Shahar and Young et al 2004], enables EPs from the several institutions collaborating in the DeGeL to create new guideline documents, thus resolving problem 4 which was the need for client application for sharing and collaborating between users (see section 2.4.4). A source guideline is uploaded into the DeGeL, and is then used by the Uruz users, EPs and KEs, to create a new *guideline document,* marked-up by the semantic labels of one of the target ontologies available in DeGeL. Uruz is sometimes used to create a guideline document *de-novo* (i.e., without using any source) by directly writing into the knowledge roles of a selected target ontology. We are currently developing an Asbru-dedicated tool to add the formal-specification level which is more graphical oriented (see chapter 6).

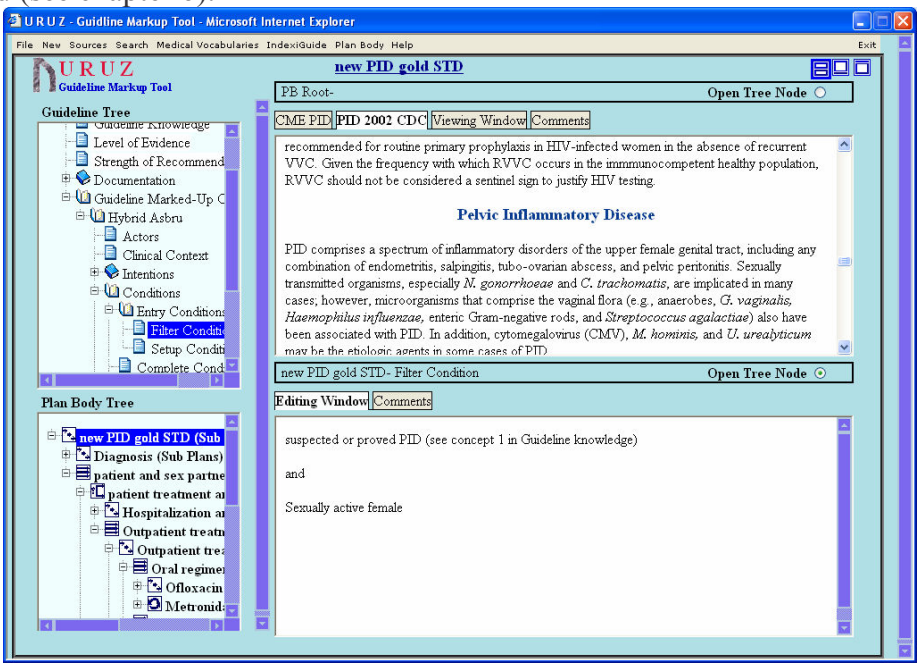


Figure 20. The Uruz Web-based guideline markup tool. The tool's basic semi-structuring interface is uniform across all guideline ontologies. The target ontology selected by the medical expert, in this case, Asbru, is displayed in the upper left tree; the guideline source is opened in the upper right frame. The expert physician highlights a portion of the source text (including tables or figures) and drags it for further modification into the bottom frame's *Editing Window* tab labeled by a semantic role chosen from the target ontology (here, the Asbru *filter condition*). Note that contents can be aggregated from different locations in the source. The *Comments* tab, the tab next to the *Editing Window* tab, stores remarks on the current selected knowledge-role, thus supporting collaboration among guideline editors.

The user of the Uruz mark-up tool browses the *source guideline* in one window, and a KR from the target ontology in the other window. To perform *semi-structured* markup, she labels the source content (text, tables, or figures) by dragging it into the knowledge-role frame, thus resolves problem 5, which was the need to use a text based sources in order to use its textual content as the base for the specification process(see section 2.4.5). Note that the editor can modify the contents or add new content. This enables turning implicit knowledge into a more explicit representation, further facilitating the task of the knowledge engineer who fully formalizes the guideline, thus resolves problem 6, which was the need for facilitating the knowledge conversion between the phases (see section 2.4.6). Since the target ontology is

selected and read on the fly (in the current implementation, as an XML file created from an XML schema), the semi-structured markup module is independent of the target ontology.

Uruz supports also adding a semi-formal Asbru representation. Semi-Formal Asbru is a simplified version of Asbru, with similar semantics to the full version, but with a somewhat less complex syntax. The main reason for using Semi-formal Asbru is to improve the collaboration between the EPs and the KEs during the GL conversion process, specifically after an EP structured the GL and before a KE converts it to Asbru. In addition, the semi-formal format still supports text-based retrieval of procedural knowledge, unlike the fully formal format. Finally, a semi-formal structure is obligatory when an electronic medical record is unavailable, since interaction with the clinical user is imperative. This property is exploited to an advantage by our hybrid runtime application module. Semi-formal Asbru has all of Asbru's knowledge-roles, such as conditions (e.g., eligibility, completion, and abort conditions), branching constructs (e.g., if-then-else or switch-case), various synchronization constraints of sub-guidelines (i.e., do in parallel, do in sequential) and time-annotations for describing temporal constraints.

Instead of using Asbru's complex notion of (plan) arguments, each GL in semi-formal Asbru has a list of patient-related data, *obtained-values,* defined during design-time. Temporal-patterns, the building blocks of a guideline in Asbru, are expressed with combinations of text and time-annotations instead of Asbru's complicated formal expressions. The semi-formal version syntax is defined using an XML schema.

A list of common clinical actions, such as drug prescription, laboratory observation and physical examination, had been added to semi-formal Asbru. These actions can be used as reusable primitive plans during guideline design-time, thus simplifying the process of GL structuring.

3.4.2. The Plan Body Wizard

To create an Asbru semi-formal representation, an Asbru-specific module, the *plan-body wizard* (*PBW*), had been embedded in Uruz (such modules can be defined also for other ontologies). The PBW is used for defining the guideline's semi-formal control structure (Figure 21). The PBW enables a user to decompose the actions embodied in the GL into atomic actions and other sub- GL s, and to define the control structure relating them (e.g., sequential actions). The PBW, used by EPs, significantly facilitates the final formal specification by the knowledge engineer. When a KEs needs to add a formal, executable, expression to a KR, she uses one of the ontology-specific Uruz modules, which delves deeper into the syntax of the target ontology. For example, in our hybrid Asbru, conditions can include temporal patterns in an expressive time-oriented query language used by all of the application modules.

To be truly *sharable*, and avoid the curly brackets problem [Tu and Musen 1999] when applying the guideline in a new environment, guidelines need to be represented in a standardized fashion. Thus, Uruz enables the user to embed in the guideline document, especially when using the PBW, terms originating from one or more of the standard, controlled vocabularies that DeGEL's vocabulary server includes, using its built-in search engine. Examples include diagnostic terms from the ICD-9-CM vocabulary, or laboratory tests from the LOINC repository (a multy-axial representation, which we are displaying to clinical users hierarchically). In all cases, the user selects a term when needed, through a uniform, hierarchical search interface to the Web-based *vocabulary server*. Thus the PBW is answering problem 2 which was facilitating the incremental specification (see section 2.4).

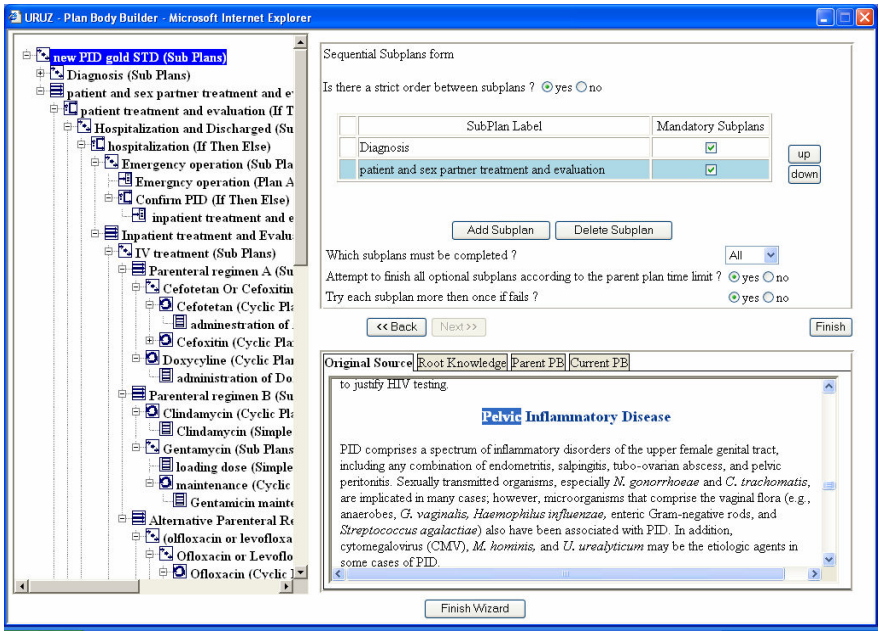


Figure 21. The Uruz Asbru semi-formal plan-body wizard (PBW) module. The module supports creation of an Asbru semi-formal control structure. On the left, the guideline's structure tree is displayed and updated dynamically as the user decomposes the guideline. On the upper right, the user is prompted with wizard-like questions to further specify the selected control structure. In the bottom right, the text of the source, current, or parent guidelines is displayed.

3.4.3. IndexiGuide: Semantic Classification of Guidelines

To facilitate guideline retrieval, the EPs indexes the GL document by one or more intermediate or leaf nodes within one or more external (indexing) semantic axes trees, using the IndexiGuide tool.(Figure 22) Currently, the semantic axes include[Shahar and Young 2004]:

1. The *Symptoms and Physical Signs* axis (e.g., hypertension), which is based on the *Medical Subject Headings* (*MeSH*) standard.
2. The *Laboratory and Special Diagnostic Procedures* axis (e.g., blood-cell counts), which is based on the CPT and LOINC standards.
3. The *Disorders* axis (e.g., endocrine disorders, neoplasms), which is based on the ICD-9 CM standard, a version of ICD.
4. The *Treatments* axis is a combination of a hierarchy of pharmacological treatments (e.g., antibiotic therapy), which is based on the *Veterans Administration NDF* (*VA-NDF*) standard, and a hierarchy of other treatments (e.g., Surgery, special therapeutic procedures, anesthesia), which is based on the CPT standard.
5. The *Body Systems and Regions* axis (e.g., pituitary gland), which is based on the MeSH standard.
6. The *Guideline Types* axis (e.g., screening, prevention, management).
7. The *Medical Specialties* axis (e.g., Genetic).

The choice of the above medical vocabularies for constructing the semantic axes was made based on a trade-off between the expressiveness of each vocabulary, and the need to represent only the top 3-4 levels of each semantic axis, which are typically sufficient for the purpose of classifying the guideline. In addition, we were looking for broadly accepted standards. For example, the LOINC medical vocabulary is not only the most expressive in its clinical domain but also the recommended one by the Health Insurance Portability and Accountability Act (HIPAA).

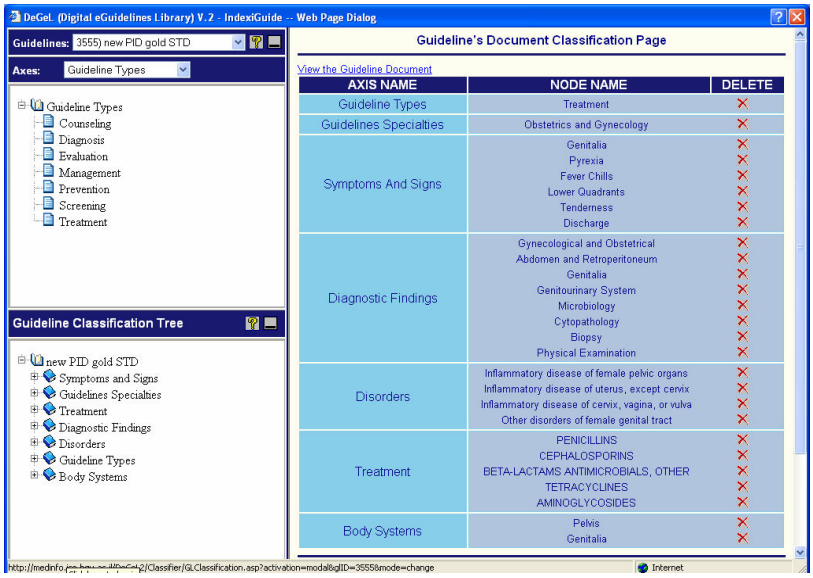


Figure 22. : The IndexiGuide and some classifications of the PID guideline. This view shows indexing by the Guideline Types axis(the left upper panel), and the other different classifications by the other axes in the right panel. The left bottom panel shown the complete tree of classification for the selected guideline document

3.4.4. Vaidurya: Context-Sensitive Search and Retrieval of Guidelines

The Vaidurya hybrid guideline search and retrieval tool [Moskovitch, Hessing et al. 2004] exploits the existence of the free-text source, the semantic indices, and the marked semi-structured-text.

Figure 23 shows the Vaidurya query interface. The user, performing a search, selects one or more concepts from one or more external semantic axes, or scopes, to limit the overall search. (e.g., disorders = hypertension). The tool also enables the user to query marked-up guidelines for the existence of terms within the internal context of one or more target-ontology's knowledge roles (e.g., in the case of Asbru, the filter condition context includes the term pregnancy).

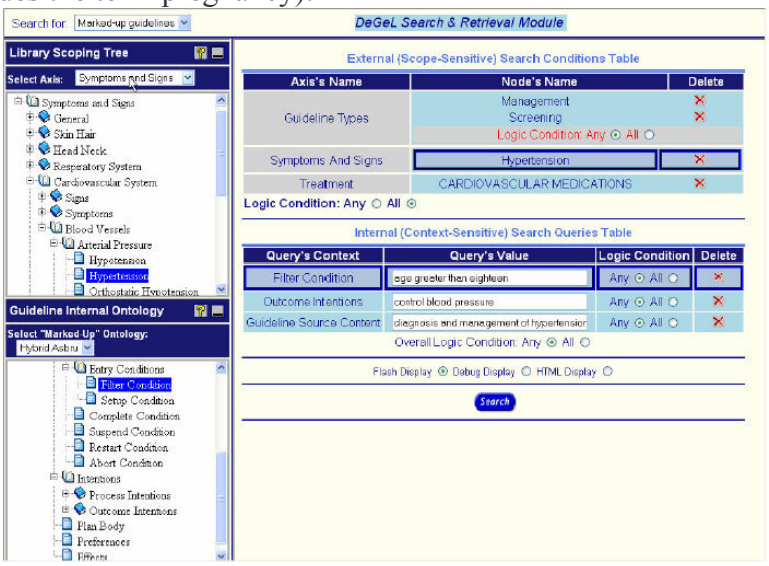


Figure 23. The Vaidurya Web-based, context-sensitive, guideline search and retrieval tool. The user defines the relevant search scope by indicating one or more nodes within the semantic axes (upper left and right frames). The search can be further refined by specifying terms to be found within the source text, and even (after selecting a target ontology such as Asbru), within the context of one or more particular knowledge roles of that ontology (middle right frame).

For search using external scopes, the default constraint is a conjunction (i.e. AND) of all selected axes (e.g., both a Cancer diagnosis within the disorders axis and a Chemotherapy therapy within the treatments axis) but a disjunction (i.e. OR) of concepts within each axis. For internal contexts, the default semantics are to search for a disjunction of the keywords within each context, as well as among contexts (i.e., either finding the term diabetes within the Filter Condition context or the term hypertension within the Effects context). The search results are browsed, both as a set and at a single-guideline level, using a specialized guideline-visualization tool.

3.4.5. The Spock Guideline Application Engine

The Spock system[Young 2005] is a runtime application of clinical guidelines designed as a *client-server* architecture to support the task of applying hybrid-Asbru clinical guidelines in a distributed and modular fashion. As a client-server system, the Spock system consists of two types of components:

1) Client side components – a Hybrid-Asbru guideline application engine, named *Spock Engine*, which encapsulates the semantics of applying a Hybrid-Asbru guideline and a graphical user interface (Figure 24) for interacting with the care provider applying the guideline.
2) Server side components – a central repository, named *Spock Server*, for logging guideline application information accumulated during intermittent guideline application sessions accompanied by a set of web-services enabling remote access to the guideline application log via the web.

In addition, the Spock system relies on additional external services for its operation. The services include:

1) Services that can retrieve the hybrid knowledge (i.e., the content of a particular knowledge-role at a specific representation level) of a guideline selected for application or its metadata (e.g., title or authors of the guideline). These services are provided by the DeGeL framework.
2) Services that can access heterogeneous clinical databases and can answer queries formulated at multiple levels of abstraction (both raw data and derived concepts). These services are provided by the IDAN architecture.
3) A service that can return the definitions of standard terms originating from controlled medical vocabularies and embedded in the guideline knowledge during its specification. These services are provided by the centralized medical vocabulary server which offers services such as search and retrieval of standard terms in numerous controlled medical vocabularies.

All of the above services are implemented using web-services, thus, enabling remote access via the web as well as interoperability capacity with other software components.

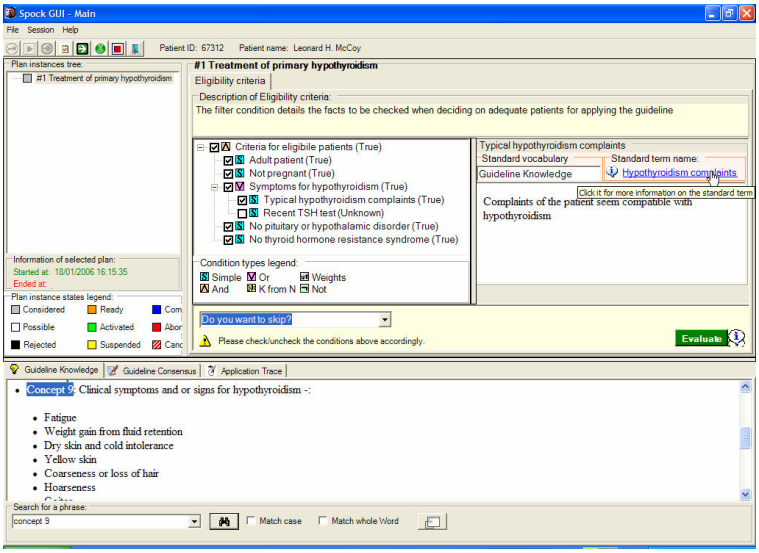


Figure 24. The main form of the Spock system's user interface after the *filter-precondition* was evaluated. Notice that when a *simple-expression's* decision criteria includes a term that is defined in the *guideline-knowledge* knowledge-role, the concept's name is displayed as a clickable link to the concept's definition displayed at the bottom panel in the *Guideline Knowledge* tab. In addition, as soon as enough simple-expressions were evaluated by the user, the Spock engine will evaluate the other composite expressions. For example, the *logical-expression* named "Symptoms for hypothyroidism" was evaluated based on the evaluation result of one of its sub-expressions named "Typical hypothyroidism complains" according to the semantics of the OR Boolean logic operator. Reproduced from [Young ,2005]

3.5. The importance of an overall methodology

As a web-based tool and part of the DeGeL framework, using the infrastructure of the hybrid guideline representation model, URUZ enables EPs and KEs on different sites to collaborate in the process of GL specification and to mark up the GL in any representation level - semi-structured, semi-formal, and formal representations - and thus solves Problems 1 to 6 (see sections 2.4.1 to 2.4.6). However, there is still a need for an overall comprehensive methodology for the GL specification process and for the evaluation of this methodology.

Bearing this in mind, the next objective of this study was to define the crucial steps for the specification process. It was known that it should involve both EPs and KEs: Patel et al. [Patel, Allen et al. 1997] have shown in GLIF that joint efforts of EPs and KEs lead to improved results. This notion is acceptable, although collaboration between EPs and KEs is considered to be the main bottleneck in the specification process, a fact which led to the recent trend of enabling the EP to perform the specification alone [Van Bemmel and Musen 1997].In addition, in their recent research Peleg et al. [Peleg, Gutnik et al. 2005] found that in the process of making algorithms from narrow guidelines, teamwork is crucial for detecting errors, and that the team should include a KE. It will be seen that the rationale of this study, that is, to find a methodology for the overall specification process, provides support for all phases before, during, and after markup, and is based on collaboration between EPs and KEs in all activities of the methodology.

The solution reached in this study for the overall markup process will be described in detail in the Methods section, which will describe our solution for Problem 7, which is the lack of an overall methodology for the specification process, and Problem 8, which is the lack of methodology for evaluation of the specification results (see section 2.4.7 and 2.4.8)

4. Methods

4.1. The overall GL Specification Methodology

In this section, a detailed methodology and description of the process of creating a consensus is presented, as well as solution for the overall process of specification.
The activities in the overall specification include three main phases: Before, during markup and after markup:

1. Before markup activities:
   - Choosing the specification language
   - Learning the specification language
   - Selecting a GL for specification
   - Create an Ontology-Specific Consensus
   - Acquiring training in the markup tool
   - Making a Gold Standard
2. During Markup activities:
   - Classifying the GL according semantic indices
   - Performing the specification process using the tools and consensus
3. After Markup activities:
   - Evaluation of the results of GL specification

The different activities are described in detail in the following subsections:

4.1.1. Choosing the specification language

The first step towards specification of GL is to select the target GL ontology for specification. As pointed previously, different ontologies can be used for different proposes: one might select for example PRODIGY for creation of common scenarios of chronic diseases, GLIF or Proforma for general GL modeling, or GEM for documentation a very comprehensive comparison between most of the methods and their representations can be found in [Peleg, Tu et al. 2002] and in [Wang, Peleg et al. 2002].However, we have used the Asbru ontology due to its expressive procedural structures and its explicit representation of the GL's time-oriented process and outcome objectives ("intentions"). Thus, we previously created a set of tools to support specification and application of Asbru-based GLs

4.1.2. Learning the specification framework and tools

The EPs are instructed by the KEs about the essential concepts and aspects required for the specification process. This activity includes learning the specification language (Asbru in our case), the hybrid model and its representation levels, the overall GL representation framework (in this case DeGeL) and its related tools. Learning the specification language will help the EPs to select a GL appropriate for formal representation considering its semantic aspects, and not only its clinical ones. A textual documents or help manual might help to support this activity. We created for this phase as for all evaluation phase a kit of documents called "Markup–Kit" which includes explanations and examples of essential aspects such as the specification language (Asbru in our case), and how to use the tools This kit will support learning, training and performing the specification process. The "Markup–Kit" and all its content as used in this research can be found in Appendix A.

4.1.3. Selecting a GL for specification

Once a decision has been made to automate a set of GLs in some clinical settings, the next step is to decide which GLs are to be formalized. A good candidate GL for specification should be one of a common disease with agreement between the majority of EPs on the methods of diagnosis and treatment, and with a clear, well defined clinical pathway. The EPs might select more then one GL source when, for example, information regarding some directive is defined in more detail in another source. These sources, in addition to their own knowledge and interpretations, will serve as the basis of knowledge for creating the consensus.

4.1.4. Creating an Ontology-Specific Consensus

- **The importance of using a consensus**

  The cognitive studies of the IntreMed group [Patel, Branch, et al. 2002] have found that, given different domain knowledge and strategies, designers and users do not represent the information in the same way, leading to different interpretations and decisions. In our experience, although there is an agreement on the GL textual source among EPs, usually there is no agreement among the EPs in the same clinical setting concerning the directives of the treatment: each EP interprets the GL differently, and thus for the same treatment different EPs have different interpretations. The problem becomes even more complicated when trying to automate the GL according to some specification language (Asbru in this case): The EPs are not programmers and thus it is hard for them to understand the semantics of the specification language; in contrast, the KEs are not familiar with the clinical semantics. Therefore, neither of them (the EP or KE) can perform the specification alone. This problem still exists when the EP has some experience with the specification language and is assigned the role of KE: some content might still be missed as the specification is then done according to only one opinion (the EP's) who thinks only unilaterally. However, as experienced as the EP can be in the specification language, he usually cannot be more experienced than a KE. Even an EP working together with a KE does not suffice, since there is no collaborative process of thinking about the specification process among all EPs in the clinical setting.

  Another problem is that the source of the GL does not supply the requirements of automation; in other words, the GL text is too abstract for automation because it can not be described by the specification language. In addition, the specification of a GL requires the EPs to answer explicitly many questions with which they do not have to contend when they use (again, with various interpretations) the GL text. Furthermore, since most of the knowledge in the GL source may be implicit rather than explicit, and sometimes the necessary knowledge in the text regarding some treatment is missing, each EP interprets the implicit or the missing content differently, leading to disagreement and a great deal of variability among the EPs. In addition, sometimes one textual source does not suffice to include all the necessary knowledge, and therefore, when specifying the GL, each EP expresses his own interpretation, a fact which leads, again, to different approaches in the treatment. Finally, it should be noted that even when there is agreement between all EPs, it has to be explicitly detailed and written.

  A uniform interpretation of the GL across the EPs, and between the EPs and the KEs, that is, a consensus, is therefore required. A consensus is defined by in this study as: "An interpretation of the GL agreed upon by the EP and the KE." In

particular, for the specification purpose, the consensus is a structural document that describes schematically the clinical directives of the GL and the semantic logic of the specification language, that is, it is an Ontology Specific Consensus (OSC). An OSC is created collaboratively by EPs and KEs, and its creation might involve converting implicit into explicit knowledge, even when it is not included in the GL sources, considering the localization of the GL and other directives that were not a part of the clinical desiderata until this step.

Attempts to create a consensus without a KE being involved in the guideline specification process led us to quite limited results. In Appendix B.1 the first attempt in this research study to create a consensus of the COPD GL using the EPs alone without involving KEs is documented. It can be seen that the declarative knowledge was defined only for the root plan (the first page), although it should have been defined for all subplans too. This happened because the EPs, having a lesser understanding of the specification language, did not know about the subplan and therefore did not define it. In addition, although the order among the schematic diagrams of the subplans was quite clear, there was some misunderstanding when it came to defining the plan's semantics, especially in the plan "Admit to Ward," the semantics of which is very complex; it can be seen that its semantics is not correct, as the *Sequential*, *If-Then Else* and *Periodic* semantic types are arranged without any logical connection between them. This might be because without the KE's input it was very difficult for the EPs to understand the *Plan Body* KR, which is the procedural knowledge, and includes semantic types such as *Cyclical* and *Sub-Plans*. After creating this consensus, the EPs tried to mark up the GL in URUZ using it. It was thus not surprising that they encountered many difficulties when they tried to structure the GL in the PBW, which led to the EPS making different markups (for the same GL), to a "dead end" in the markup process, and to the involvement of the KEs. The KEs tried to solve the consensus problems, but it was quite clear that the collaboration and standardization of the guideline scenarios should be done again from scratch **before** the markup process and include all clinical and semantic aspects of the GL in great detail. Furthermore, it was found that the creation of a consensus regarding the semantics of the GL is an indispensable, crucial mandatory step before markup, and should be done in the first stage by a group of EPs from the medical field, in collaboration with a KE who is familiar with the specification language. The later stages of creating an OSC can be done by a senior EP who has a lot of practical knowledge and experience and a KE. Peleg et al. [Peleg, Gutnik et al. 2005] suggest that to avoid errors in encoding guidelines it should be verified that: 1) all relevant knowledge of the guideline is carried to the algorithm; 2) all the necessary information to rank the treatment is provided; and 3) patient scenarios are considered. It will be seen that making an ontology specific consensus supports all these issues

- **Methodology for creating an ontology-specific consensus**

The consensus is created in a systematic iterative fashion by performing the following steps:

1. After selecting the GL source (see Figure 25 for example of the textual source for the "inpatient treatment of the PID GL), the EPs in the clinical institution where the GL will be applied is performing "brain storming" in order to create a clinical consensus based on the GL sources. The consensus can sometimes be modified with a little refinement of common local clinical settings as well as by adding concepts and directives by the EPs, representing their own knowledge and expertise.
   This clinical consensus should describe, in flow chart fashion, the general clinical steps of the GL; each step is described in a rectangular task box with a suitable title. Decisions are intuitive described as rhombus. When one step needs to be divided further into sub-steps, an arrow is added, pointing to the next task box. This clinical consensus describes the basic procedural knowledge of the GL, and is free of ontology-specific details (Figure 26).
2. The KE and the EPs collaborate. The KE, who is an expert in the ontology specification language, together with the senior EP overviews the clinical consensus and contributes his emphasis regarding the structure of the clinical consensus, adding some control structure in the notion of the specification language. For example, he might define the order for a group of steps, or refine the text and the logic of the clinical consensus, according to the answers the EP gives to his questions. The result is that a clinical consensus is refined, that is, the ontology-specific *procedural* part of the consensus (Figures 27, 28).
3. The EPs and the KE create a table of the relevant KRs of the specification language for each defined step (task box) which they decide is necessary (note that not all the steps will necessarily be detailed here), and describe the content of each KR. In this step, the input of the KE is very important: he must revise the semantics of each step, and look for contradictions or special cases where the semantics of the ontology-specific language is not clear to the EP, and is therefore not defined properly. The result is the ontology-specific *declarative* part of the consensus (Table 4).

After some iterations of steps 3 and 4 , an **Ontology-Specific Consensus (OSC)** is formed, composed of the two parts (see Appendix B for all consensuses documents): 1) **Procedural** - describes the structure of the steps composing the GL (which are mainly derived from the clinical consensus) 2) **Declarative** - describes for each step its properties according to the semantic KRs of the specification language. The following table summarizes the steps in the consensus methodology:

Table 4. Summery of steps towards consensus

Step	The Participant	The process	The Output of this process
1	The EPs of the clinical institution where the GL will be applied in	Describing in flow chart (in task box) fashion the general clinical steps of the GL	Clinical consensus
2	The KEs, who familiar with the specification language together with the EPs	Adding control structure in the notion of the specification language for each step (e.g. parallel, define if the plan is mandatory for application)	The *Procedural* part of the clinical consensus refined with ontology-specific language.
3	The KEs with the EPs	Creating a table of the relevant KRs of the specification language for each defined step (task box) which they decide is necessary	The *Declarative* part of the clinical consensus refined with ontology-specific language.
4	The knowledge engineer together with the expert physicians	After some iterations of steps 3 and 4 , an Ontology-Specific Consensus (OSC) is formed	Ontology-Specific Consensus (OSC) , composed of the two parts: procedural and declerative

Parenteral Regimen A

Cefotetan 2 g IV every 12 hours

OR
Cefoxitin 2 g IV every 6 hours
PLUS
Doxycycline 100 mg orally or IV every 12 hours.

Parenteral Regimen B

Clindamycin 900 mg IV every 8 hours

PLUS
Gentamicin loading dose IV or IM (2 mg/kg of body weight) followed by a maintenance dose (1.5 mg/kg) every 8 hours. Single daily dosing may be substituted.

Alternative Parenteral Regimens

Ofloxacin 400 mg IV every 12 hours

OR
Levofloxacin 500 mg IV once daily
WITH or WITHOUT
Metronidazole 500 mg IV every 8 hours
OR
Ampicillin/Sulbactam 3 g IV every 6 hours
PLUS
Doxycycline 100 mg orally or IV every 12 hours.

Figure 25.: The textual representation of the "Inpatient treatment of PID" plan. The text shown is taken verbatim from the guideline source text. The complete textual source can be found in Appendix B.2.

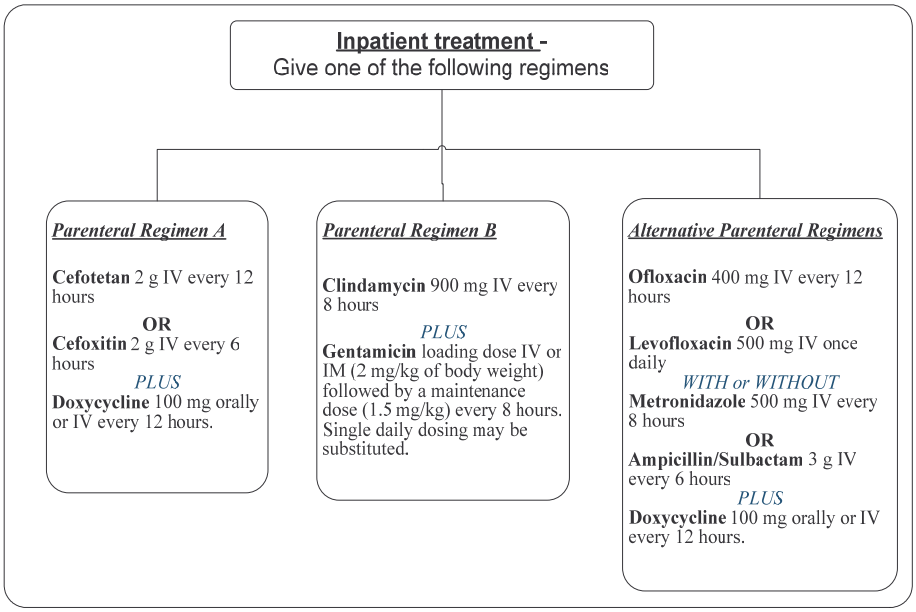


Figure 26. The first stage in forming a consensus: the task boxes are defining the clinical pathway of choosing between three regimens, in this case for the GL "Inpatient treatment of PID".

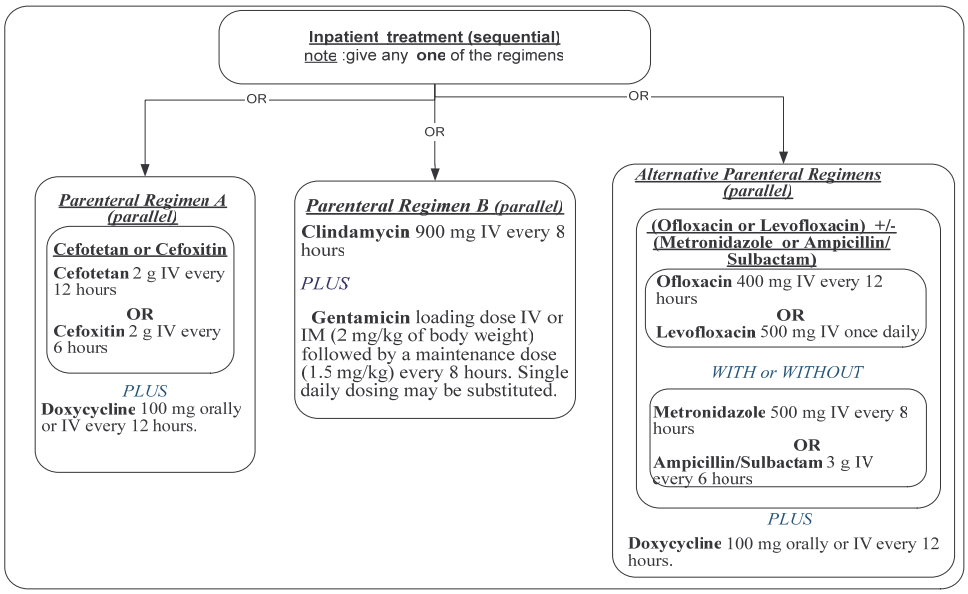


Figure 27. The second stage in forming an ontology specific consensus, in this case, for "Inpatient treatment and evaluation" plan of the "Pelvic Inflammatory Disease" guideline. Each procedural step is a task box. Note the semantic order in each task box (e.g., parallel or sequential) the knowledge engineer add after defining the clinical consensus.

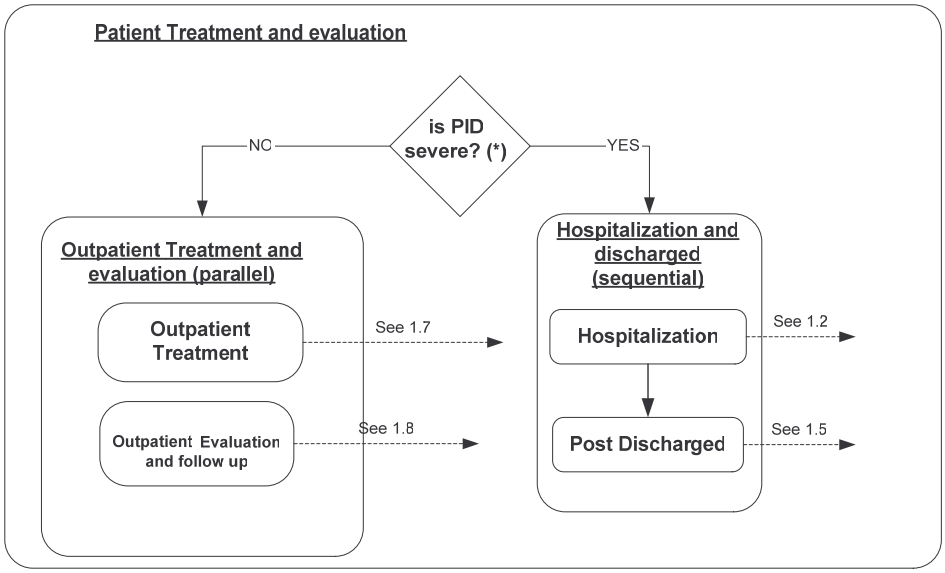


Figure 28. The second stage in forming an ontology specific consensus, in this case, for "patient treatment and evaluation" plan of the "Pelvic Inflammatory Disease" guideline. Each procedural step is a task box. Note the semantic order in each task box (e.g., parallel or sequential) the knowledge engineer add after defining the clinical consensus. Note also the arrows, pointing to the next sub-steps for further structuring.

Table 5. The third stage in forming a consensus.

IV treatment	
Level of evidence	
Strength of recommendation	
Actors	Doctors – gynecological ward
Clinical context	Ward
Intention *intermediate process*	To administer IV drug according to one of the regimens A,B, Alternative
Intention *overall outcome*	To achieve substantial clinical improvement (*) **OR** To allow PO treatment
Filter Condition	
Set Up Condition	
Abort Condition	
Complete Condition	Substantial clinical improvement (*) **OR** Post IV PO treatment started

This table shows the results of defining its declarative part, in this case we use Asbru knowledge roles to describe the declarative part of "IV treatment" plan. Note the * notation which implies a concept (substantial clinical improvement in this case) that the editor should elicit as part of the guideline knowledge

4.1.5. Training the EP's in markup tool

The EPs who intend to perform the mark-ups are instructed by the KEs in the specification tool (URUZ markup tool in our case) and in the OSC. A help manual or small simulation of marking up another GL may be used to assist the EP in this process of refreshing the meaning of some concepts or to confirm understanding about a task related to the specification language, the specification tool, or generally about the specification process. We used the "Markup-Kit" as explained in 4.1.2, which can be found in Appendix A

4.1.6. Creating the Gold Standard Markup

For each of the GLs, a Gold Standard (GS) is created by a senior EP and a KE together, using the OSC and GL source. The GS considered to be the best markup and most detailed from both the clinical and semantic aspects, and therefore is used only for evaluation of the markups. This step can be performed before or in parallel to the activity of performing the markups.

4.1.7. Performing the markup

After the EP feels enough confident, he/she can start to specify the GL using the markup tool (URUZ, in our case) according to the OSC and the GL sources. In addition, the EP classifies the GL according to a set of semantic indices (e.g. diagnosis, treatment) using the IndexiGuide Tool (see section 3.4.3). However, the KE is not part of this session, and may help the EP in case of technical problems..

4.1.8. Evaluation of the markups

After the markups are completed, they are evaluated by comparing them to the GS according to objective measures. Evaluation of the markup is important because it helps to qualify its quality in qualitative and quantitative measures. This stage is done by a senior EP and a KE using a designated evaluation tool which enables scoring those measures for each plan, sub-plan and KR of evaluated markup and described in detail in detail in section 4.3 -4.6. Our implementation for evaluation of the markups is described in detailed in section 4.2

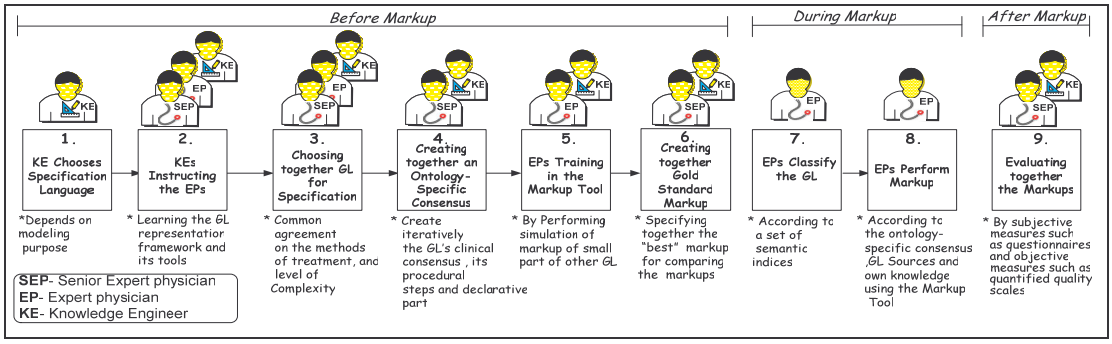


Figure 29. The three main phases of the methodology before, during and after the markup, and the activities in each phase. Note the descriptions under each activity. Activity six (creation of a gold standard) can be performed in start, or in parallel with activities seven and eight (editors markup). Note also the participants in each activity.

Summary:

A methodology was designed for the specification process. It was implemented using the DeGeL framework and its related tools; in particular The URUZ Markup Tool was used to facilitate the markup process (all tools are described in detail in chapter 3.4). It was necessary to design and implement evaluation methods in order to measure and quantify the quality of the elicited knowledge both subjectively and objectively.

This definition of a comprehensive overall methodology for performing markups resolves problem 7: lack of an overall methodology for the specification process (see section 2.4). The solution for the last problem 8, which is the lack of methodology for evaluation of the specification (see section 2.4) is describe in detail in the next section.

4.2. The Evaluation Design

Recalling the common problems encountered in the evaluation of KA tools, and the unresolved problem no.8, i.e., lack of an evaluation methodology (see sections 2.4.8), and keeping in our mind the conclusions from pervious KA evaluation (see section 2.3.3), helped us to design the evaluation , to put-forward and considered some specific criteria:

1. Amount of Expertise for this evaluation
2. The acquired knowledge domain we will use for the evaluation
3. The Ontology Specific Consensuses for each GL in the evaluation
4. The Gold Standard markup for each GL
5. The Markups for each GL
6. The Evaluation of markups

Those criteria are described in detail in the following subsections:

4.2.1. Amount of Expertise

As explained in section 2.4, finding large enough samples of experts to give the evaluation statistical significance is difficult. It is more difficult to find EPs who are specialize and are not, for example, general family medicine experts. It is yet more difficult to find specialist EPs who have sufficiently computer-orientation to be able to use complicated task such as performing markup using URUZ.

Despite these problems, eight EPs and two KEs were enlisted. Of the EPs, some participated in the creating the OSC and GS phases, and some participated in making the markup itself, all for the purposes of the evaluation, as described in the following table:

Table 6. The different participants in the evaluation For each participant it describes his type (expert physician or knowledge engineer), his academic belonging and place of work, his position and denotes a short label

Type	Academic Belongs to	Works at	Specializes in	Denoted as
Expert Physician (EP)	Faculty of Health Sciences at Ben-Gurion University	Soroka Medical Center of Ben-Gurion University	The head of ob/gyn division	EP1
			Head of gyn Emergancy room (ER)	EP2
			Resident at ob/gyn division	EP3
			Senior Endocrinologist in the internal medicine department	EP4
			Intern in the internal medicine department	EP5
	Faculty of Health Sciences at Tel-aviv University	E. Wolfson Medical Center	Resident at Otolaryngology-Head and Neck Surgery	EP6
	Stanford University Medical School, Department of Medicine	The Veterans Affairs (VA) Palo Alto Health Care System (PAHCS)	Practicing clinician in internal Medicine, Pulmonary diseases and critical care and a major teaching affiliate of Stanford	EP7
			Center for Primary Care & Outcomes Research, Stanford University	EP8
Knowledge Engineer(KE)	Information system department in the engineering faculty, at Ben-Gurion University	The Medical Informatics Research Center	Medical informatics engineer	KE1
			Medical informatics engineer	KE2

4.2.2. The acquired knowledge domain

Since three medical experts in three different clinical domains had been enlisted for the purpose of the evaluation, it was decided to select three guidelines from each domain, in incremental level of detail:

- *Treatment of pelvic inflammatory disease (PID)* – the guideline's domain is *gynecology* and it is intended for use by gynecologists to treat patients suffering from an inflammatory disease related to the pelvis [PID-CDC, PID-Emedicine 2005]
- *Treatment of Chronic Obstructive Pulmonary Disease (COPD)* – the guideline's domain is *pulmonology* and it is intended for use by emergency department physicians and/or general medical ward physicians to treat patients suffering from a low respiratory rate related to chronic obstruction of the pulmonary system. [COPD 2005]
- *Treatment of primary hypothyroidism (HypoThyrd)* – the guideline's domain is *endocrinology* and it is intended for use by family practitioners to treat patients suffering from hypothyroidism that is directly related to the thyroid gland (i.e., primary) [HYPERTHYROIDISM-AACE 2002]

4.2.3. The Ontology Specific Consensuses

For each of the guidelines, an OSC was formed in increasing level of detail, through the collaboration of an EP and a KE:

1. **PID** – The PID OSC was made by EP1, EP3 and KE1. It was decided to add, in addition to the CDC source, another source to effect a better description of the diagnosis (although later it was decided not to evaluate this branch). Other refinements were related to inpatient and outpatient evaluations, and treatment of sex partners. The OSC starts with some recommended diagnostic measures of the PID disease (e.g., ultrasound), continues with determining the severity of the patient's PID state (e.g., severe or mild), recommends whether the patient should be hospitalized or not (i.e., inpatient or outpatient options), suggests appropriate treatment options (e.g., a particular drug regimen), and schedules an evaluation follow-up of the patient's condition. (For the PID OSC see document Appendix B.3.).
   In order to test the ability of the EPs to formalize expressions and plans, it was decided not to specify in the OSC declarative part the operators between the expressions (such as "And","Or" operators), and not to explicitly define what are the mandatory and optional plans in the OSC
2. **COPD** – The COPD OSC was made by EP7, EP8, KE1 and KE2. The OSC was tightly related to the sources, and actually contains all of its content, maybe because the source is very well organized, structured and divided by sections, which facilitates converting it into steps. The OSC starts with an initial treatment (e.g., mechanical ventilation), continues with evaluation and diagnosis of the patient's respiratory state (e.g., chest radiograph), and recommends that the patient be either transferred to the Intensive Care Unit (ICU) or admitted to a general medical ward to continue treatment. (For initial OSC document see Appendix B.4. for its final document see Appendix B.5.). Again, in order to test the ability of the EPs to formalize expressions and structures, it was decided, this time to include explicitly and to define the

operators between the expressions (such as "And","Or"), but still not to explicitly define what are the mandatory and optional plans in the OSC

3. **Hypothyroidism** – The Hypothyroidism OSC was made by EP4 and KE2. In this case, the knowledge was in most cases was EP4's own knowledge. The OSC starts with recommended diagnosis actions of the disease (e.g., TSH measurements), continues with determining the severity of the patient's hypothyroidism state (e.g., unequivocal or subclinical), recommends the initial treatment (e.g., initial dose of the TSH replacement drug named *Levothyroxine*), and finally enters a repeated loop, probably for the patient's lifetime, of monitoring the patient's state (e.g., repeated measurements of TSH values) and corresponding treatment adjustment (e.g., change dose of recommended drug) guideline. (For the OSC document see Appendix B.6.).
However, for this GL, it was decided, to explicitly define the operators between the expressions (such as "And","Or"), and to explicitly define what are the mandatory and optional plans in the OSC.

4.2.4. The Gold Standard

For each of the GLs, a GS was made together by an EP and a KE. The GS will be used to evaluate the markups, and is assumed to be the best markup and also the most correct in its clinical and Asbru aspects. In the case of the PID GL, the GS was made by EP3 and KE1. The EPs and the KEs who specified the GS used a pre-defined OSC, which was the same OSC that the EPs used for markup. In the case of the COPD, EP7 and KE1 prepared the GS, and in the case of the HypoThyrd GL, EP6 and KE2.

4.2.5. The Markups

For each GL, two markups were made by the EPs. Each EP, after training in the use of URUZ, used the Markup Kit (see Appendix A), the OSC, the GL sources, and his own knowledge. EP1 and EP2 performed markup for the PID; EP5 and EP8 performed the markups for the COPD and HypoThyrd. In the case of the PID, the EPs performed the markups in their own "playground", that is, in their clinical setting, sometimes between their "real" job of treating patients. In addition, in each markup, the EP classified the GL using the IndexiGuide.

4.2.6. The evaluation of markups

As a result of lessons learned regarding other evaluations and the problems which arose (see section 2.3.2. and 2.4), we decided to conduct a very comprehensive evaluation to quantify the knowledge using two types of measure:

- **Subjective Measures**
  Questionnaires were administered to find the concepts which most helped the EPs in the process of making the GS, and their attitude regarding the specification language and its associated tools; a usability test was also administered. (See section 4.3.)
- **Objective Measures**
  In order to obtain meaningful quantitative results, it was decided to measure the markup outputs on two scales: completeness of the markup and correctness of the markup in two aspects: clinical correctness and Asbru semantics correctness.

Finally, it is important to clarify that this objective measurement is achieved by comparing the markups with the GS. (See section 4.3.)

The evaluation process is done through the collaboration of an EP and a KE, using the Evaluation Tool (see section 4.5): the PID evaluation was done by EP1 and KE1, the COPD evaluation by EP7 and KE1, and the HypoThyrd evaluation by EP4 and KE1.

Summary

The following table summarizes all the stages of the evaluation: for each evaluated GL, it describes its clinical domain, the participants in stage of creating OSC, in the stage in making the GS, the EPs who performed the markups, and finally the participants in the evaluation sessions.

Table 7. The participant in each phase of the evaluation.

Guideline	Clinical Domain	Ontology Specific Consensus made by	Gold Standard made by	Markups made by	Evaluation made by
PID	Gynecology	EP1, EP3 and KE1	EP3 , KE1	EP1 , EP2	EP1,KE1
COPD	Pulmonology	EP7,EP8, KE1 and KE2	EP7 , KE1	EP5 ,EP8	EP7,KE1
Hypothyroidism	Hypothyroidism	EP4 , KE2	EP5 , KE2	EP5 ,EP8	EP4,KE1

Note that EP1 and EP8 who participated in the stage of making an ontology specific consensus, were also participates in the stage of creating the markups for the evaluation Not also that EP5 and EP8 marked up both COPD and the HypoThyrd GLs

4.3. The Subjective Measures

In the scope of the subjective measures, we wanted to know which aspects helped the EPs in creating the OSC and which aspects helped them in performing the markups. We also wanted to know how the EPs understand Asbru KRs before and after performing the markup, and to know what the usability of the URUZ markup Tool is. To answer all those issues, we administered the following three groups of questioners:

4.3.1. Questioners regarding the aspects helped the EPs' in creating an OSC and making the markups

21 aspects, which were thought to be essential to know in the process of making the OSC and the markup, were listed. For each concept, a description and an example was given. Thus, two questionnaires were composed (for the questionnaires documents see appendix C.1 and C.2). The first questionnaire was administered after the OSC was formed, and the second after the markup. In addition, the correlation, if any, between those two questionnaires was examined. The next two sections summarize those two questionnaires:

1. **Questionnaire 1: The EPs' attitude regarding making an OSC**
   - Purpose - to check the EPs attitude regarding making an OSC
   - Target participants - EPs who participated in the process of making the OSC
   - Instructions – "Give a grade for each of the following concepts, expressing in what way it helped you to make an ontological-specific consensus."
   - Grade: [-3 interfered a lot / 3 contributed a lot]
   - For Questionnaire 1 see appendix C.1

2. **Questionnaire 2: The EPs' attitude regarding making a markup**
   - Purpose - to check the EP's attitude regarding the different aspects they used during the markup process
   - Target participants - all EPs who performed the markup
   - Instructions – "In what way did each of the following concepts help you when performing markup?"
   - Grade : [-3 interfered a lot / 3 contributed a lot]
   - For Questionnaire 2 see appendix C.2

4.3.2. Questioners regarding the EPs' understanding Asbru KRs and specify them

Another interesting issue was to evaluate the EPs attitude regarding Asbru KRs, and the difficulty of structuring them. To answer this issue, the EPs were given a list of all the Asbru KRs and were asked through two different questionnaires to grade them before and after performing the markup (for the questionnaires documents see appendix C.3 and C.4). In addition, the correlation, if any, between those two questionnaires was examined. The list of Asbru KRs can be found in section 3.4.1**.** The next two sections summarizes those two questionnaires

3. **Questionnaire 3: The EPs attitude regarding Asbru KRs before markup**
   - Purpose - to check the EP's attitude regarding their understanding of Asbru KRs
   - Target participants - all EPs who performed the markup
   - Instructions – "How difficult was it for you to understand each one Asbru KRs before performing markup?"
   - Grading : [-3 difficult to understand / 3 very easy to understand]
   - For Questionnaire 3 see appendix C.

4. **Questionnaire 4: The EPs' attitude regarding Asbru KRs after markup**
   - Purpose - to check the EP's attitude regarding structuring Asbru KRs
   - Target participants - all EPs who performed the markup
   - Instructions – "How difficult was it for you to structure each one of Asbru KRs while performing markup?"
   - Grades : [-3 difficult to structure / 3 very easy to structure]
   - For Questionnaire 4 see appendix C.4

4.3.3. Usability Test for URUZ

Finally, for testing URUZ's usability using a standard test, we used the Scalable usability test (SUS) [Brooke 1996]. This test was deemed very important to us because its results would influence the next version of the specification tool. (for the SUS questionnaire see appendix C.5.)

- Purpose - to check URUZ usability
- Target participants - all subjects
- Instructions – "For each question fill the most appropriate SUS grade."

Summary

The results of these questionnaires contributed to the researcher's understanding of the attitude of the EPs regarding the concepts in the methodology, the specification language, and the Markup Tool. Such insights are very important with regard to drawing qualitative conclusions about future work. In addition, objective measures were used.

4.4. The Objective Measures

4.4.1. Scales of objective measures

Two main scales to quantify the quality of markups were defined:
1.**Completeness**:

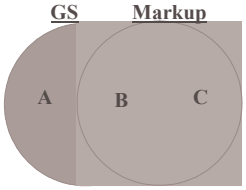


Figure 30. A graphical show of the different existence groups: content missing in the markup and exist in the gold standard belongs to group A; content exist in the markup and exist in the gold standard belongs to group B; content exist in the markup and missing in the gold standard belongs to group C.

When comparing the markup to the GS, most of the content of the GS should ideally be included in the markups, that is, a high level of *completeness* should be expected (Group B in Figure 30). However, some mistakes can be made, such as the EPs skipping some content. In this case, content is missing from the markups, and therefore the level of completeness might be lower (Group A in Figure 30). Another possible situation is that EPs may add to the markup some content which does not exist in the GS (Group C in Figure 30); in this case the level of completeness will not be lower, since it was decided, for simplicity's sake, to define completeness as being the case where all content which exists in the GS exists in the markup (Groups A and B together in Figure 30).
Thus, we can define Group A as the "Missing" group, Group B as the "Existing" group, and Group C as the "Redundant" group (see Table 8).

Table 8. The three existence groups and their name and descriptions

Group	Name	Description
A	Missing	Content exists in the GS and missing in the Markup
B	Existing	Content exists in the GS and exists in the Markup
C	Redundant	Content missing in the GS and exists in the Markup

2.Correctness
We defined two measures of correctness to quantify the quality of the elicited content of the matkups:

- **Clinical Measure (CM)** – measure the clinical correctness of the content
- **Asbru Semantic Measure (ASM**) - measure the semantics correctness of the content (Asbru semantic in our case)

This scale is used for every KR in each Plan. Usually the CM is assigned by the EP, and the ASM by the KE. For each measure, the score was defined according to a quality scale of measure. Table 9 presents the quality scale measures and the possible scores which can be assigned to an EP's markup content in each of the two measures. Note that the score is always assigned for both correctness measures (CM and ASM) by comparing the content of the markup of an EP to the GS. Thus, a uniform measure for the different markups, specified by different EPs markup of the same GL is assured.

Table 9. The quality scale of clinical and Asbru semantics measures. Note that the score is always given for both of measures (CM and ASM).

Scale of measure	Assigned Score	Description
Clinical (CM)	1	Correct clinically
	0	Incorrect clinically, without worsening the patient's outcome
	-1	Incorrect clinically, and worsening the patient's outcome
Asbru Semantic (ASM)	1	Correct according to Asbru semantics
	0	Incorrect according to Asbru semantics without worsening the patient's outcome
	-1	Incorrect according to Asbru semantics, and worsening the patient's outcome

4.4.2. The types of errors in the evaluation

Two main types of errors an EP can make during markup were identified: general errors which are general for all KRs, and specific errors which are specific errors in a specific KR type. The types of error are detailed in the following sections:

1. **General errors:**
   We define the general errors in two scales:
   1.1. Clinical errors:
   - Clinical content not accurate
   - Clinical semantics not well specified
   - Clinical content not complete.

   1.2. Asbru semantics errors:
   - Asbru semantics content not accurate
   - Asbru semantics content not well specified
   - The content does not includes mappings to standard terms
   - The necessary knowledge is not defined in the guideline knowledge when it should be.

2. **Specific errors:**
   For each KR Type a specific error was identified:
   2.1. Conditions /Intensions KRs:
   - There are no And/Or operators between the different criteria.

   2.2. Simple Action Plan-Body Type:
   - Has no text content describing the plan
   - Has no single atomic action semantics with clear specification and description for the action to be performed.

   2.3. Plan Activation Plan-Body Type:
   - Plan name is not defined
   - Defined plan does not exist in DeGeL.

   2.4. Cyclical Plan-Body Type:
   - Has no text content describing the plan
   - Has no semantics of repeating action
   - "Repeating specification" content is missing
   - Has no semantics of "starting specification"
   - "Starting specification" is missing

- Has no semantics of "frequency duration"
- "Frequency duration" is missing.

2.5. If-Then-Else Plan-Body Type:
- Has no text in PB
- Condition content not semantics of condition
- No And/Or operators between the different criteria in condition
- Has no clear specification for the condition
- Has no definition for "then" case
- "Then" case has no semantics of action
- Has no definition for "else" case when GS states that one should exist.
- "Else" case has no semantics of action.

2.6. Switch case Plan-Body Type:
- Has no criteria in the switch clause at all
- The switch clause criteria do not have appropriate semantics
- Criteria have no multiple values
- Some or all alternative steps are not further defined
- Does not have at least 2 possibility values
- The steps are not mutually exclusive.

2.7. Sub-plans Plan-Body Type:
- Has no text content describing the plan
- Not all the necessary subplans are further defined
- At least 2 subplans do not exist
- The plans are not given an order (i.e., parallel, sequential)
- The defined order is wrong
- There is no specification for all "hard" elements: mandatory, max attends, waiting strategy, completed spec. when there should be.

Thus, the number of each type of error which each EP performed qualitatively can be measured.

4.4.3. Resolution of Measure

There are three different GLs and two markups for each GL. Each markup is composed of plans, and each plan has a set of KRs. Thus, there are three major cross-sections of completeness and correctness to measure:

- GLs - to find common trends in a GL, and in all GLs
- EPs - to find trends in between the markups of the EPs across the same GL and between GLs
- KRs - to find trends in a specific KR type and common trends across KRs and KR classes across one markup, GL and in all GLs

Thus, some more specific levels of the resolutions for the completeness and correctness measures can be defined:

1.1. A Single KR in a plan - Single KR in a plan, for example the *Filter Condition* KR

1.2. A Plan: - Plan with one or more KRs, but at least plan-body KR

1.3. KR type: Mean measure of aggregated KRs with the same type in the markup, for example, all Actors KRs in the markup.

1.4. KR Class:

To allow abstraction, classes of common KRs with a common spirit were defined. The following table describes the different KR classes: *Context*, *Intentions*, *Conditions* and *Plan-Body*, and the relevant KRs in each class (see Table 10):

Thus, the completeness and correctness can be measured not only for each KR alone or aggregated, but for a class of KRs. Thus, trends or common repeated problems in a class can be identified, and it is then possible to draw conclusions. For example, it may be found that *Plan-body* KR class constantly gets a lower grade than the *Conditions* KR class, or that *Intentions* KR class has a substantially higher level of completeness.

Table 10. The KR classes and the relevant KRs in each class

Class	Relevant KRs in class
Context	Actors
	Clinical context
Intentions	Overall - outcome
	Overall - process
	Intermediate Outcome
	intermediate - process
Conditions	Filter Condition
	Abort Condition
	Setup condition
	Suspend condition
	Reactivate condition
	Complete Condition
Plan-Body	Simple Action
	If-then -else
	Repeating plan
	Subplans – parallel order
	Subplans – sequential order
	Plan-activation
	Switch case
	To be defined

1.5. Tasks:

Different KR classes create different contexts of tasks: for purposes of **Application** of the GL, only the KR classes of *Condition* and *Plan-Body* are required, whereas, for Quality Assurance (**QA**) task the *Context, Intentions and Plan-Body* KRs classes are required. Thus, we can define two tasks: the *Application* task and the *QA* task. The following table describes the different tasks and the relevant KR class for each of them:

Table 11. The different tasks and their relevant KR classes

Task	Relevant KRs classes
Application	Conditions
	Plan-Body
Quality Assurance	Context
	Conditions
	Plan-Body

1.6. Overall Markup : This is the highest resolution, which is the mean measure of all KR classes in a markup, that is a markup of an EP

1.7. GL – The average scores of all markups of the same GL, and all GLs

Summary

Using the resolutions above, the completeness and correctness can be measured across the main incisions axes of measures: GL, EP, and KR.

Each resolution can be calculated across of KR type, KR class, a markup of EP, across EPs in the same GL or between GLs, and for all GLs. Thus , we can define the formulas to calculate the completeness score and the Mean Quality Score(MQS) of each resolution.

4.4.4. Measures of completeness

After defining the different resolutions of measurements, we can define the measures for completeness for each KR, KR class, markup of an EP, a GL (mean of the markups of its EPs editors), and for all GLs, using the following cross-sections:

- KR – denoted here as index *i*
- EP – denoted here as index *j*
- GL – denoted here as index *k*

1.Completeness for a Single KR:

Recalling the existence groups (see 4.4.1), a single KR *i* marked up by an EP *j* in a GL *k* can exist in one of the following groups (KR_g *ijk)*: the *Missing* group if it is exists in the GS and is missing in the Markup, *Existing* group if it exists in the GS and exists in the markup, or the *Redundant* group if it is missing in the GS and exists in the Markup.

Thus, its completeness can be defined by:

(1) $\mathbf{KR_g}$ ***ijk*** - KR *i* in the marked up of EP *j* in GL *k* in existence group *g*.

Example:

The KR *Filter Condition* (*i*) in the marked of EP1(*j*) in the GL of PID(*k*) is missing (compared to the GS), and therefore belongs to the *Missing* group (*g*), or existing in the markup and therefore belongs to the *Existing* group (*g*).

2.Completeness for a specified Plan :

A plan belong s to the *Existing* group if it has a Plan-Body KR. If the plan doesn't have a Plan-Body KR(which means the EP didn't specified it), it belongs to the *Missing* group. Actually, the existence of a plan is a private case of completeness of a single KR measure (see 1), with the KR *Plan-Body* (KR_g *i =Plan-Body jk).* Thus, similarly to KR, the completeness of each plan *i* in the markup of an EP *j* in a GL *k* can be measured according its belonging to each one of the existence groups (P_g *ijk*).

Thus, the completeness can be defined by:

(2) $\mathbf{P_g}$ ***ijk*** – Plan *i* in the markup of EP *j* in GL *k* in existence group *g*

Example

The plan "IV treatment"(*i)* in the markup of EP1(*j*) in the PID GL (*k*) is missing (compared to the GS), and therefore belongs to the *Missing* group (*g*), or existing in the markup and therefore belongs to the *Existing* group (*g*).

Thus, the measure for completeness for the number of specified plans in a markup of EP *j* in GL *k* in existence group *g* (NP_g *jk*) is defined:

(3) $\mathbf{NP}_{g \exists \{\text{"missing group"},\text{"existing group"},\text{"redundant group"}\}}\ \mathbf{jk} = \sum P_g\ ijk$.

Example

In the markup of EP1 (*j*) in the PID GL (k), there are 10 missed plans, 80 existing plans, and 3 redundant plans. Thus, the completeness for the specified plans for this markup will be:

$NP_{g \exists \text{"mis sin g"}} = 10$; $NP_{g \exists \text{"exisitng"}} = 90$; $NP_{g \exists \text{"redundant"}} = 10$

Since the total number of plans in the *Missing* group and the number of plans in the *Existing* group in each markup in a GL is always the same as the number of plan in the GS for this GL, the measure for the total number of plans in each markup of an EP in a GL *k* (NPG *k*) can be defined:

NPG *k* $= NP_{g \exists "missing"} jk + NP_{g \exists "exisitng"} jk$

Example

Following previous example, the total number of plans in each markup of EP in the PID GL (*k*) is : NPG *PID = 10+ 90=100*

Note that although EP1 is defined here, the total number of plans is the same for all markups of EPs in the PID GL.

Thus, the *completeness* for an EP *j* in a GL *k* in each existence group *g* ($EPC_g jk$) can be defined:

(4) $\mathbf{EPC}_{g \exists \{"missing\ group","existing\ group","redundant\ group"\}}$ **jk %** $= \frac{NP_g\ jk}{NPG\ k}$.

Example

Following example 2 and 3, the completeness of EP1(*j*) in the PID GL(*k)* is*:*

$EPC1_{g \exists \{"missing\ "\}}$ PID % $= \frac{10}{100} = 10\%$

$EPC1_{g \exists \{"existing"\}}$ PID % $= \frac{90}{100} = 90\%$

$EPC1_{g \exists \{"redundant"\}}$ PID % $= \frac{10}{100} = 10\%$

Note that the completeness measure of the *Redundant* group is in proposition to the NPG *k* measure (which describes how many redundant plans there are in relation to the GS). This is because a common denominator of all measures was required. Thus, for each markup its completeness (in proposition to the GS) is measured in each existence group.

In addition, the *completeness* for a GL *k*, in each existence group *g* (GLC_g *k)*, can be defined assuming *n* EPs in the GL *k*:

(5) $\mathbf{GLC}_{g \exists \{"missing\ group","existing\ group","redundant\ group"\}}$ **k %** $= \frac{\sum_{j=o}^{j=n} NP_g\ jk}{n}$

Example

There are 2 EPs (*j*) in the PID GL(*k)*, and for the existing group (*g)* each of them have the following measure of completeness:

$EP1_{g \exists \{"existing"\}}$ PID % = 90% ; $EP2_{g \exists \{"existing"\}}$ PID % = 80%

Thus, the completeness of the PID GL(*k*) in the existing group (*g) is* :

$GLC_{g \exists \{"existing\ group"\}}$ *PID* % $= \frac{90+80}{2} = 85\%$

Finally, the *completeness* for all GLs, in each existence group *g*, assuming *n* GLs ($GLCs_g$) can be defined:

(6) $GLCs_{g \exists \{"missing\ group","existing\ group","redundant\ group"\}}$ **%** $= \frac{\sum_{k=o}^{k=n} NPG\ k\ \ * GLC_g\ k}{\sum_{k=0}^{k=n} NPG\ k}$.

Example

There the 3 GLs (*k*) with the following measure for the *Existing* group:

NPG *PID* =106 ; NPG *COPD* =53 ; NPG *HypoThyrd* =39

$GLC_{g \exists \{"existing\ group"\}}\ PID\ \% = 85\%$; $GLC_{g \exists \{"existing\ group"\}}\ COPD\ \% = 90\%$

$GLC_{g \exists \{"existing\ group"\}}\ HypoThyrd\ \% = 95\%$

Thus, the *completeness* measure for all GLs is in the *Existing* group:

$$GLCs_{g \exists \{"existing\ group"\}} = \frac{106*85\% + 53*90\% + 39*95\%}{106+53+39} = 80.5\%$$

3.Completeness for a KR Type:

The number in each set of single KRs (see formula 1 in section 1) from the same type *i* with *n* KRs in markup EP *j* of GL *k* in existence group g ($NK_g\ ijk$) can be defined:

(7) $\mathbf{NK}_{g \exists \{"\text{missing group}","existing\ group","redundant\ group"\}}\ \boldsymbol{ijk} = n * K_g\ ijk$

Example:

$K_{g \exists \{"existing\ group"\}}\ ijk$ - is the *Filter Condition* (*i)* KR in markup of EP1 (*j)* in the PID GL (*k*) in the *Existing* group (*g*).

Suppose in this markup of EP1, there are 50 instances of *Filter Condition* KR in the *Existing* group and he missed 10 KRs, that is, belong to the *Missing* group, and he have 5 KRs in the *Redundant* group. Thus, the number in this markup is in each group is:

$NK_{g \exists \{"existing\ group"\}}$ *Filter-condition EP1 PID =50*

$NK_{g \exists \{"missing\ group"\}}$ *Filter-condition EP1 PID =5*

$NK_{g \exists \{"redundant\ group"\}}$ *Filter-condition EP1 PID =5*

Since the total number of KRs in the *Missing* group plus the number of KRs in the *Existing* group in each markup in a GL is always the same as the total number of KRs in the GS for a GL, the measure for the total number of KRs of type *i* in each markup of EP *j* in GL *k* (NKG *ik*), can be defined:

(8) $\mathbf{NKG}\ \boldsymbol{ik} = NK_{g \exists \{"missing\ group"\}}\ ijk + NK_{g \exists \{"existing\ group"\}}\ ijk$

Example

Following example 8, the total number of *Filter Condition* KR (*i)* in the PID GL (*k)* is:

NKG *Filter Condition PID* = 50+5 =55

Then, the *completeness* for each KR of type *i* in markup EP *j* of GL *k* in existence group *g* ($KRC_g\ ijk$) can be defined:

(9) $\mathbf{KRC}_{g \exists \{"\text{missing group}","existing\ group","redundant\ group"\}}\ \boldsymbol{ijk}\ \boldsymbol{\%} = \frac{NK_g\ ijk}{NKG\ ik}$

Example:

Following examples 8 and 9, the completeness of the *Filter Condition* KR (*i)* in the markup of EP1 (*j*) in the PID GL (*k)* is:

$KRC_{g \exists \{"existing\ group"\}}\ ijk\ \% = \frac{50}{55} = 91\%$

$KRC_{g \exists \{"missing\ group"\}}\ ijk\ \% = \frac{5}{55} = 9\%$

$KRC_{g \exists \{"redundant\ group"\}}\ ijk\ \% = \frac{5}{55} = 9\%$

Then, the *completeness* for each KR of type *i* for a GL *k*, in each existence group *g* ($KRGC_g\ ik$) , assuming *n* EPs is defined:

(10) $\mathbf{KRGC}_{g\exists\{\text{"missing group","existing group","redundant group"}\}}\ \boldsymbol{ik}\ \mathbf{\%} = \dfrac{\sum_{j=o}^{j=n} \mathrm{KRC}_{g}\ \mathrm{ijk}}{n}$

Example:

Suppose there are 2 EPs (*j*) in the PID GL(*k*), and for the *Existing* group (*g)* each of them have the following measure of completeness for the *Filter Condition* KR (*i)*:

$\mathrm{KRC}_{g\exists\{\text{"existing"}\}}$ *EP1 Filter Condition PID* % = 91%;

$\mathrm{KRC}_{g\exists\{\text{"existing"}\}}$ *EP*2 *Filter Condition PID* % = 80%;

Thus the completeness of the PID GL(*k*) in the existing group (*g) is* :

$$\mathrm{KRGC}_{g\exists\{\text{"existing group"}\}}\ PID\ \% = \frac{91+80}{2} = 85.5\%$$

Then, the *completeness* for each KR *i* for all GLs, in each existence group *g* (KRCi), assuming *n* GLs can be defined:

(11) $\mathbf{KRC}_{g\exists\{\text{"missing group","existing group","redundant group"}\}}\ \boldsymbol{i}\ \mathbf{\%} = \dfrac{\sum_{k=o}^{k=n} \mathrm{NKG}\ ik\ * \mathrm{KRC}_{g}\ ik}{\sum_{k=0}^{k=n} \mathrm{NKG}\ ik}$

Example:

There the 3 GLs (*k*) with the following measure for the *Filter Condition* KR (*i*) in the *Existing* group (*g*):

NKG *PID* =50 ; NKG *COPD* =30 ; NKG *HypoThyrd* =20

$\mathrm{KRGC}_{g\exists\{\text{"existing group"}\}}$ *PID* % = 85.5% ; $\mathrm{KRGC}_{g\exists\{\text{"existing group"}\}}$ *COPD* % = 90%

$\mathrm{KRGC}_{g\exists\{\text{"existing group"}\}}$ *HypoThyrd* % = 95%

Thus, the *completeness* measure for all GLs is in the *Existing* group:

$$\mathrm{KRC}_{g\exists\{\text{"existing group"}\}}\ \textit{Filter Condition} = \frac{50*85.5\% + 30*90\% + 20*95\%}{50+30+20} = 88.75\%$$

4.Completeness for KR Class:

After the completeness measure for each KR type has been obtained, the weighted measure for a KR class can be defined.The completeness using a weighted mean has to be calculated because each KR *i* has different NK_{g} *ijk* (different numbers of KRs).

Thus, the *completeness* for each KR class *c* with group of *i* KRs in markup of EP *j* in GL *k* in existents group *g* ($\mathrm{KCC}_{g}cjk$) can be defined, assuming *n* KRs in each KR class, assuming *n* KR types in a KR class :

(12) $\mathbf{KCC}_{g\exists\{\text{"missing group","existing group","redundant group"}\}}\ \boldsymbol{cjk}\ \mathbf{\%} = \dfrac{\sum_{i=o}^{i=n} \mathrm{KRC}_{g}\ ijk\ * \mathrm{NKG}\ ik}{\sum_{i=0}^{i=n} \mathrm{NKG}\ \mathrm{ik}}$

Example:

If we have 40 KRs (NKG *ik*) of the type *Actors* with completeness (KRC_{g} *ijk)* of 90%, and 30 KRs (NKG *ik*) of the type *Clinical Context* with completeness (KRC_{g} *ijk)* of 95% in the markup of EP1(j) in the PID GL in the Existing group, then the completeness measure for the Context KR class is :

$$\mathrm{KCC}_{g\exists\{\text{"existing group"}\}}\ \textit{Context EP1 PID} = \frac{40*90\% + 30*95\%}{40+30} = 92\%$$

Then, the *completeness* for each class *c* with group of KRs for the GL *k*, in each existence group *g* (KCC_g *ck)* can be defined assuming *n* EPs:

(13) $\mathbf{KCC}_{g \exists \{\text{"missing group","existing group","redundant group"}\}}$ ***ck* %** = $\frac{\sum_{j=o}^{j=n} KCC_g \text{ cjk}}{n}$.

Example:
EP1 (*j*) achieved in the *Context*(*c*) in the *Existing* group (*g*) in the PID GL(*k*) a completeness (KC_g *ck*) of 92%. EP2 achieved for the same KR class in the same GL a completeness (KC_g *ck*) of 95% , thus the completeness in the PID GL of the *Context* KR class in the Existing group is :

$$KCC_{g \exists \{\text{"existing group"}\}} \ Context\ PID = \frac{92\% + 95\%}{2} = 93.5\%$$

The number of a set of KRs in each class *c* in the GS of a GL *k* (NKG *ck*) can be defined, assuming *n* KR types in the class:

(14) **NKG *ck*** $= \sum_{i=0}^{i=n} NKG\ ik$.

Example:
If we have 40 instances (NKG *ik*) of the *Actors* KR type, and 30 instances (NKG *ik*) of the *Clinical Context* KR type in the PID GL(*k*), then the total number in the *Context* KR Class of the PID GL is :

NKG *Context* PID = 30 + 40 =70

Then, the *completeness* for each class *c* with set of KRs *i* for all GLs, in each existence group *g* (KCC_g *c*) can be defined, assuming *n* GLs :

(15) $\mathbf{KCC}_{g \exists \{\text{"missing group","existing group","redundant group"}\}}$ ***c* %** = $\frac{\sum_{k=o}^{k=n} KCC_g\ ck \ * NKG\ ck}{\sum_{k=0}^{k=n} NKG\ ck}$

Example:
The completeness in the PID GL(*k*) of the *Context* KR class(*c*) in the *Existing* group (*g*) (KCC_g *ck*) is 90% with 50 KRs (NKG *ck*). The completeness in the COPD GL is 95% with 40 KRs and in the PID GL the completeness is 85% with 30 KRs. Thus the completeness of the *Context* KR class for all GLs is:

$$KCC_{g \exists \{\text{"existing group"}\}} \ Context\ \% = \frac{90\% * 50 + 95\% * 40 + 85\% * 30}{50 + 40 + 30} = 90.4\%$$

5.Completeness for Task:

The completeness of a task *t* is the weighted mean of its classes; Thus, the *completeness* for each task *t* which is composed of classes *c* in markup of EP *j* in GL *k* in existence group *g* can be defined (KTg *tjk)*, assuming *n* classes in each task:

(16) $\mathbf{KTC}_{g \exists \{\text{"missing group","existing group","redundant group"}\}}$ ***tjk* %** = $\frac{\sum_{c=o}^{c=n} KCC_g\ cjk \ * NKG\ ck}{\sum_{c=0}^{c=n} NKG\ ck}$

Example:
If we have 50 KRs (NKG *ck*) of the *Conditions* KR class with completeness (KCC_g *cjk)* of 90%, and 80 KRs (NKG *ck*) of the *Plan-Body* KR class with completeness (KCC_g *cjk)* of 95% in the markup of EP1(j) in the PID GL in the *Existing* group, then the completeness measure for the *Application* task is :

$$\mathrm{KTC}_{g\ \exists\{\text{"existing group"}\}}\ \textit{Application EP1 PID} = \frac{40*90\% + 30*95\%}{40+30} = 92\%$$

Then, the completeness for each task *t* with group of classes *c* for GL *k*, in each existence group *g* can be defined, assuming *n* EPs:

$$\mathrm{KTC}_{g\ \exists\{\text{"missing group","existing group","redundant group"}\}}\ tk\ \% = \frac{\sum_{j=o}^{j=n} \mathrm{KTC}_g\ tjk}{n}\ .$$

Example:
EP1 (*j*) achieved in the *Application* task(*t*) in the *Existing* group (*g*) in the PID GL(*k*) a completeness ($\mathrm{KTC}_g\ tk$) of 92%. EP2 achieved for the same task in the same GL a completeness ($\mathrm{KTC}_g\ tk$) of 95% , thus the completeness in the PID GL of the *Application* task in the *Existing* group is :

$$\mathrm{KTC}_{g\ \exists\{\text{"existing group"}\}}\ \textit{Application PID} = \frac{92\% + 95\%}{2} = 93.5\%$$

The number of KRs in each task *t* in the GS of a GL *k* can be defined, assuming *n* classes in each task *t*:

(17) $\mathbf{NKG}\ tk = \sum_{c=0}^{c=n} \mathrm{NKG\ ck}$.

Example:
If we have 40 KRs (NKG *ck*) of in the *Conditions* KR class, and 30 KRs (NKG *ck*) of the *Plan-Body* KR class in the PID GL(*k*), then the total number of KRs in the *Application* task in the PID GL is :
NKG *Application* PID = 30 + 40 =70

Then, the *completeness* for each task *t* with group of classes *c* for all GLs *k*, in each existence group *g* can be defined, assuming *n* GLs:

(18) $\mathbf{KTC}_{g\ \exists\{\text{"missing group","existing group","redundant group"}\}}\ \boldsymbol{t\%} = \frac{\sum_{k=o}^{k=n} \mathrm{KTC}_g\ tk\ * \mathrm{NKG}\ tk}{\sum_{k=0}^{k=n} \mathrm{NKG}\ tk}$.

Example:
The completeness in the PID GL(*k*) of the *Application* task (t) in the *Existing* group (*g*) ($\mathrm{KTC}_g\ tk$) is 90% with 100 KRs (NKG *tk*). The completeness in the

$$\mathrm{KTC}_{g\ \exists\{\text{"existing group"}\}}\ \textit{Application}\ \% = \frac{90\%*50 + 95\%*40 + 85\%*30}{50+40+30} = 90.4\%$$

6.Completeness of the overall Markup:

The completeness of a markup is the weighted mean of its classes; Thus, the *completeness* for each markup which is composed of classes *c* in markup of EP *j* in GL *k* in existence group *g* can be defined (MCg *jk)*, assuming *n* classes in each markup:

(19) $\mathbf{MC}_{g\ \exists\{\text{"missing group","existing group","redundant group"}\}}\ \boldsymbol{mjk\ \%} = \frac{\sum_{c=o}^{c=n} \mathrm{KCC}_g\ cjk\ * \mathrm{NKG}\ ck}{\sum_{c=0}^{c=n} \mathrm{NKG}\ ck}$

Example:
If we have 50 KRs (NKG *ck*) of the *Conditions* KR class with completeness ($\mathrm{KCC}_g\ cjk$) of 90%, and 70 KRs (NKG *ck*) of the *Context* KR class with completeness of 95% , and 80 KRs of the *Intentions* KR class with completeness

of 100% ,and 80 KRs of the *Plan-Body* KR class with completeness of 95% in the markup of EP1(j) in the PID GL in the *Existing* group, then the completeness measure for the overall markup is :

$$MC_{g \exists \{"existing\ group"\}} EP1\ PID = \frac{50*90\% + 70*95\% + 80*100\% + 80*98\%}{50+70+80+80} = 96.3\%$$

Then, the completeness for GL *k*, in each existence group *g* (GC_g *k*) can be defined, assuming *n* EPs in the GL:

$$GC_{g \exists \{"missing\ group","existing\ group","redundant\ group"\}}\ k\ \% = \frac{\sum_{j=o}^{j=n} MC_g\ jk}{n} .$$

Example:
EP1 (*j*) achieved in his markup (*m*) in the *Existing* group (*g*) in the PID GL(*k*) a completeness (KMC_g *mjk*) of 92%. EP2 achieved for the same task in the same GL in his markup a completeness of 95% , thus the completeness in the PID GL of the two markups in the *Existing* group is :

$$GC_{g \exists \{"existing\ group"\}}\ PID = \frac{92\% + 95\%}{2} = 93.5\%$$

The number of KRs in each markup of a GL *k* can be defined, assuming *n* classes in each markup *m* (NKG *k*):

(20) $$\mathbf{NKG}\ \boldsymbol{k} = \sum_{c=0}^{c=n} NKG\ ck .$$

Example:
If There are 40 KRs (NKG *ck*) of in the *Context* KR class, and 50 KRs (NKG *ck*) of in the *Intentions* KR class, and 30 KRs (NKG *ck*) of in the *Conditions* KR class, and 40 KRs (NKG *ck*) of the *Plan-Body* KR class in the PID GL(*k*), then the total number of KRs in the PID GL is :

NKG PID = 40+50+30 + 40 =160

Then, the *completeness* for all GLs in each existence group *g* can be defined, assuming *n* GLs:

(21) $$\mathbf{GC}_{g \exists \{"missing\ group","existing\ group","redundant\ group"\}}\ \mathbf{\%} = \frac{\sum_{k=o}^{k=n} GC_g\ k\ * NG\ k}{\sum_{k=0}^{k=n} NG\ k} .$$

Example:
The completeness in the PID GL(*k*) in the *Existing* group (*g*) (GC_g *k*) is 90% with 50 KRs (NKG *k*), The completeness in the COPD GL(*k*) in the *Existing* group (*g*) (GC_g *k*) is 95% with 40 KRs (NKG *k*) and the completeness in the HypoThyrd GL(*k*) in the *Existing* group (*g*) (GC_g *k*) is 85% with 30 KRs (NKG *k*)... The completeness of all GL s is

$$\mathbf{GC}_{g \exists \{"existing\ group"\}} = \frac{90\%*50 + 95\%*40 + 85\%*30}{50+40+30} = 90.4\%$$

4.4.5. Measures of Correctness

Considering the two measures for correctness: *Clinical* measure (CM) and *Asbru* measure (ASM) (see section 2 in 4.4.1), we can define the following measures for each KR, KR class, markup of an EP, a GL (mean of the markups of its EPs editors), and for all GLs (Note that the correctness is measure for all existences group), using the following incisions:

- KR – denoted here as index i
- EP – denoted here as index j
- GL – denoted here as index k

1.Correctness for a single KR:

Since CM and ASM have the same weight, the *correctness* measure for a single KR i in markup of EP j in GL k (SKC ijk) can be defined:

(22) $$\textbf{SKC } ijk = \frac{CM + ASM}{2} .$$

Example:

The *Filter Condition* KR (i) in some plan in the markup of EP1 (j) in the PID GL (k) assigned in the evaluation session the scores of CM = 1, and ASM = 0. Thus, its correctness measure is:

$$\text{SKC } \textit{Filter-Condition EP1 PID} = \frac{1+0}{2} = 0.5$$

2.Correctness for a plan:

Plan can have one or more KRs, thus the *correctness* measure for each plan with n KRs in markup of EP j in GL k (PC ijk)can be defined:

(23) $$\textbf{PC jk} = \frac{\sum_{i=1}^{i=n} \text{SKC } ijk}{n} .$$

Example:

The plan "IV treatment" in the in the markup of EP1 (j) in the PID GL (k) have three KRs with the following correctness measures (SKC ijk): *Filter-Condition* KR with SKC of 0.5, *Abort-Condition* KR with SKC of 0.5, and *Plan-Body* KR with SKC of 0, thus the correctness of the "IV treatment" plan is:

$$\text{PC } \textit{"IV treatment" EP1 PID} = \frac{0.5+0.5+0}{3} = 0.66$$

3.Correctness for a KR Type:

Each KR is evaluated according to Clinical and Asbru semantics measures, and assigned to one of these values:[1, 0,-1]. Thus, if all KRs are aggregated with the same type in each markup the following divisions can be made:

Table 12. The different amount of marked up KRs of the same type categorized by its scale and its score in each scale

Scale	-1	0	1	Total
Clinical scale	Num of KRs with clinical score -1 $NKC_{s=-1}ijk$	Num of KRs with clinical score 0 $NKC_{s=0}ijk$	Num of KRs with clinical score 1 $NKC_{s=1}ijk$	TKC ijk = $\sum_{s=0,-1,1} NKC\ (s)_{ijk}$
Asbru scale	Num of KRs with Asbru score -1 $NKA_{s=-1}ijk$	Num of KRs with Asbru score 0 $NKA_{s=0}ijk$	Num of KRs with Asbru score 1 $NKA_{s=1}ijk$	TKA ijk = $\sum_{s=0,-1,1} NKA\ (s)_{ijk}$

Thus, the total number of scores instances of both measures for a KR *i* in markup of EP *j* in GL *k* (TKS_{ijk}) can be defined:

(24) $\mathbf{TKS_{ijk}} = TKC_{ijk} + \mathrm{TKA_{ijk}}$.

Example:

There are total of 120 instances in the Clinical measure and 120 instances in the Asbru measure of the *Filter Condition* KR (*i*) of markup of EP2 (*j*) in the PID GL (*k*), thus its total number of scores instances(TKS*ijk*) is :

TKC *Filter Condition EP2 PID = 120+120=240*

Then, we can calculate the proportion of scores of 1 for the KR *i* in the markup of the EP *j* the GL *k* ($PK_{s=1\ ijk}$) can be defined:

(25) $\mathbf{PK_{s=1}\ ijk\ \%} = \dfrac{NKC_{s=1}ijk + NKA_{s=1}ijk}{TKS\,ijk}$

Example:

Following the previous example, There are 50 instances of KRs with clinical score of 1 ($NKC_{s=1}ijk$) and 40 instances of KRs with Asbru score of 1($NKA_{s=1}ijk$) for the *Filter Condition* KR (*i*) of markup of EP2 (*j*) in the PID GL (*k*), thus its proportion of scores of 1 ($PK_{s=1}\ ijk\ \%$) is:

$\mathrm{PK_{s=1}}$ *Filter Condition EP2 PID %* $= \dfrac{50+50}{240} = 41.7\%$

Thus, the Clinical Mean Quality Score of correctness for KR *i* in markup of EP *j* in GL *k* (CMQS_{ijk}) can be defined:

(26) $\mathbf{CMQS}_{ijk} = \dfrac{\sum_{s=0,1,-1} NKC(s)_{ijk} * s}{\mathrm{TKC_{ijk}}}$.

Example:

In the clinical measure of the *Filter Condition* KR (*i*) of markup of EP2 (*j*) in the PID GL (*k*) there are 40 instances of scale (*s*) -1, 30 instances of scale (*s*) 0, and 50 instances of scale(*s*) 1 , thus its CMQS_{ijk} will be:

CMQS *Filter Condition EP2 PID* $= \dfrac{(40*-1)+(30*0)+(50*1)}{40+30+50} = 0.083$

In the same way, the Asbru Mean Quality Score of correctness for KR *i* in markup of EP *j* in GL *k* (AMQS_{ijk}) can be defined:

(27) $\mathbf{AMQS}_{ijk} = \dfrac{\sum_{s=0,1,-1} NKA(s)_{ijk} * s}{\mathrm{TKA_{ijk}}}$

And the Mean Quality Score (MQS) for KR *i* in markup of EP *j* in GL *k* can be defined:

(28) $\mathbf{MQS}_{ijk} = \dfrac{CMQS_{ijk} * TKC_{ijk} + AMQS_{ijk} * TKA_{ijk}}{\mathrm{TKC_{ijk}}}$.

Example:

Following the pervious examples, the clinical measure (CMQS) of the *Filter Condition* KR (*i*) of markup of EP2 (*j*) in the PID GL (*k*) is 0.083 with total of 120 instances (TKC). Supposed that the Asbru measure (AMQS) is 0.5 with total of 120 instances (TKA) , thus its MQS is:

MQS Filter Condition EP2 PID $= \dfrac{0.083*120+0.5*120}{240} = 0.29$

Thus, the MQS for each KR Type can be measured, thus revealing, for example, if one EP achieved lower MQS for filter condition KR than for abort condition.
The total number instances of KR *i* in GL *k* (TGS_{ik}) assuming n EPs can be defined:

(29) $\mathbf{TGS_{ik}} = \sum_{j=0}^{j=n} TKS_{ijk}$

The mean MQS for a KR *i* for a GL *k* ($GMQS_{ik}$) assuming *n* EPs can be defined:

(30) $\mathbf{GMQS_{ik}} = \dfrac{\sum_{j=o}^{j=n} MQS_{ijk} * TKS_{ijk}}{TGS_{ik}}$

Example:
In the PID GL (*k*) for the *Filter Condition* KR(*i*) EP1(*J1)* achieved MQS of 0.5 with 70 total instances scores (TKS), and EP2 (*J2)* achieved MQS of 0.29 with 120 total instances scores(TKS), thus the PID GL MQS will be :

$$\text{GMQS } \textit{Filter Condition PID} = \frac{0.5*70+0.29*120}{70+120} = 0.36$$

The same formula can be used for calculating the MQS of KR *i* for all GLs ($GsMQS_i$), assuming m GLs:

(31) $\mathbf{GsMQS}_i = \dfrac{\sum_{k=0}^{m=k} GMQS_{ik} * TGS_{ik}}{\sum_{k=0}^{m=k} TGS_{ik}}$

Example:
The PID GL (*k1*) has an GMQS in the Filter Condition KR (*i)* of 0.3 with TGS of 190 instances (see previous example). In the COPD GL (*k1*) for the same KR there is 0.5 GMQS with 60 TGS and the HypoThyrd(*k1*) 0.65 and 80 TGS instances. Thus the MQS for the *Filter Condition* in all GLs is:

$$\text{GsMQS } \textit{Filter Condition} = \frac{0.3*190+0.5*60+0.65*80}{190+60+80} = 0.42$$

Note that the same formulas can be used for each of the PB Types, by calculating for each PB type its CMQS, AMQS and its MQS

4.Correctness for a KR Class:
The MQS for a KR Class can be calculated by the weighted mean MQS of each of its KRs. Thus, we can define for KR Class *c* in markup of EP *j* in GL *k* the total number KR instances (TKC*cjk*), assuming *n* KRs in the class:

(32) $\mathbf{TKC}cjk = \sum_{i=0}^{i=n} TKSijk$.

Thus, the MQS of KR Class *c* in markup of EP *j* in GL *k* ($CLMQS_{cjk}$) can be defined assuming *n* KRs in the class

(33) $\mathbf{CLMQS_{cjk}} = \dfrac{\sum_{i=0}^{i=n} MQS_{ijk} * TKS_{jik}}{TKC_{cjk}}$

Example:
In the PID GL (*k*) for the markup of EP1 (*j*), the *Context* KR Class (*c*) containing the *Actors* KR (*i1*) with MQS of 0.7 and TKS of 40 and the *Clinical Context* KR (*i*2) with MQS of 0.8 and TKS of 50.

Thus, the MQS for the *Context* KR Class is:

CLMQS *Context* EP1 PID = $\frac{0.7*40+0.8*50}{40+50} = 0.75$

The total number instances of KR Class c in GL *k* (TCS_{ck}) assuming n EPs can be defined:

(34) $\mathbf{TCS_{ck}} = \sum_{j=0}^{j=n} TKC_{cjk}$

Thus, the mean MQS for a KR Class *c* for a GL *k* ($GCMQS_{ck}$) assuming *n* EPs in the GL can be defined:

(35) $\mathbf{GCMQS_{ck}} = \frac{\sum_{j=o}^{j=n} CLMQS_{cjk} * TKC_{cjk}}{TCS_{ck}}$

Example:

In the PID GL (*k*) for the *Context* KR Class(c) EP1(*J1)* achieved CLMQS of 0.5 with 70 total instances scores(TKC), and EP2 (*J2)* achieved CLMQS of 0.29 with 120 total instances scores(TKC), thus the PID GL GCMQS will be :

GCMQS *Context PID* = $\frac{0.5*70+0.29*120}{70+120} = 0.36$

The same formula can be used for calculating the MQS of KR Class c for all GLs ($GCsMQS_c$), assuming m GLs:

(36) $\mathbf{GCsMQS_c} = \frac{\sum_{k=0}^{m=k} GCMQS_{ck} * TCS_{ck}}{\sum_{k=0}^{m=k} TCS_{ck}}$

Example:

The PID GL (*k1*) has a GCMQS in the *Context* KR Class(c*)* of 0.3 with TCS of 190 instances (see previous example). In the COPD GL (*k1*) for the same KR Class there is 0.5 GCMQS with 60 TCS and the HypoThyrd (*k1*) 0.65 and 80 TCS instances. Thus the MQS for the *Context* KR class in all GLs is:

$GCsMQS_c$ *Context* = $\frac{0.3*190+0.5*60+0.65*80}{190+60+80} = 0.42$

5.Correctness for a Task:

The MQS for a task can be calculated as the weighted mean MQS of each of its KR classes ($CLMQS_{cjk}$), and thus the correctness for a Task *t* can be defined:

To start with , the total number of its KR instances across the KR classes (TT*tjk*), assuming *n* KR Classes s in the task can be defined:

(37) $\mathbf{TT_{tjk}} = \sum_{c=0}^{c=n} TKC_{cjk}$.

Thus, the MQS of Task t in markup of EP *j* in GL *k* ($TMQS_{tjk}$) can be defined assuming *n* KR Classes in the task

(38) $\mathbf{TMQS_{tjk}} = \frac{\sum_{c=0}^{c=n} CLMQS_{cjk} * TKC_{cjk}}{TT_{tjk}}$

Example:

In the PID GL (*k*) for the markup of EP1 (*j*), the *Application* task (*t*) containing the *Conditions* KR class (*c1*) with CLMQS of 0.7 and TKC of 40 and the *Plan-Body* KR class (*c*2) with CLMQS of 0.8 and TKC of 50.

Thus, the MQS for the *Application* task is:

$$\textbf{TMQS}\ \textit{Application}\ \text{EP1 PID} = \frac{0.7*40+0.8*50}{40+50} = 0.75$$

The total number instances of KRs in Task *t* in GL *k* (TTS_{tk}) assuming n EPs in a GL can be defined:

$$(39)\quad \mathbf{TTS_{tk}} = \sum_{j=0}^{j=n} \mathrm{TT_{tjk}}$$

Thus, the mean MQS for a Task *t* for a GL *k* ($GTMQS_{tk}$) assuming *n* EPs in the a GL can be defined:

$$(40)\quad \mathbf{GTMQS_{tk}} = \frac{\sum_{j=o}^{j=n} TMQS_{tjk} * TT_{tjk}}{TTS_{jk}}$$

Example:

In the PID GL (*k*) for the *Application* task (*t*) EP1(*J1)* achieved TMQS of 0.5 with 70 total instances scores(TT), and EP2 (*J2)* achieved TMQS of 0.29 with 120 total instances scores(TT), thus the PID GL GTMQS will be :

$$\text{GTMQS}\ \textit{Application PID} = \frac{0.5*70+0.29*120}{70+120} = 0.36$$

The same formula can be used for calculating the MQS of Task t for all GLs ($GsTMQS_t$), assuming m GLs:

$$(41)\quad \mathbf{GsTMQS_t} = \frac{\sum_{k=0}^{m=k} TMQS_{tk} * TTS_{tk}}{\sum_{k=0}^{m=k} TTS_{tk}}$$

Example:

The PID GL(*k1*) has an GTMQS in the *Application task* (*t)*of 0.3 with TTS of 190 instances (see previous example). In the COPD GL (*k1*) for the same task there is 0.5 GTMQS with 60 TTS and the HypoThyrd(*k1*) 0.65 and 80 TTS instances. Thus the MQS for the *Application task* in all GLs is:

$$\text{GsTMQS}\ \textit{Application} = \frac{0.3*190+0.5*60+0.65*80}{190+60+80} = 0.42$$

6.Correctness for the overall Markup

The MQS for a markup can be calculated as the weighted mean MQS of all of its KR classes ($CLMQS_{cjk}$), and thus it is possible to define:

We can define for a markup of EP *j* in GL *k* the total number of its KR instances across the KR classes (TM*jk*), assuming *n* KR Classes s in the markup

$$(42)\quad \mathbf{TM}\boldsymbol{jk} = \sum_{c=0}^{c=n} \mathrm{TKC_{cjk}}\quad .$$

Thus, the MQS of markup of EP *j* in GL *k* ($MMQS_{jk}$) can be defined assuming *n* KR Classes in the markup

$$(43)\quad \mathbf{MMQS_{tjk}} = \frac{\sum_{c=0}^{c=n} CLMQS_{cjk} * TKC_{cjk}}{TM_{jk}}$$

Example:

In the PID GL (*k*) for the markup of EP1 (*j*) containing the *Conditions* KR class (*c1*) with CLMQS of 0.7 and TKC of 40 and the *Plan-Body* KR class (*c*2) with CLMQS of 0.8 and TKC of 50, *Context* KR class (*c3*) with CLMQS of 0.7 and TKC of 40 and the *Intentions* KR class (*c*2) with CLMQS of 0.8 and TKC of 50.

Thus, the MQS for his markup is:

$$\text{MMQS } EP1\ PID = \frac{0.7*40 + 0.8*50 + 0.7*40 + 0.8*50}{40+50+40+50} = 0.75$$

The total number instances of KRs of all markups in GL *k* (TMS_k) assuming n EPs in a GL can be defined:

(44) $$\mathbf{TMS_k} = \sum_{j=0}^{j=n} \mathrm{TM}_{jk}$$

Thus, the mean MQS for a GL *k* ($GMMQS_k$) assuming *n* EPs in the a GL can be defined:

(45) $$\mathbf{GMMQS_k} = \frac{\sum_{j=o}^{j=n} MMQS_{jk} * \mathrm{TM}_{jk}}{\mathrm{TMS}_k}$$

Example:

In the PID GL (*k*), EP1(*J1)* achieved MMQS of 0.5 with 70 total instances scores(TM), and EP2 (*J2)* achieved MMQS of 0.29 with 120 total instances scores(TM), thus the PID GL GMMQS will be :

$$\text{GMMQS } PID = \frac{0.5*70 + 0.29*120}{70+120} = 0.36$$

The same formula can be used for calculating the MQS for all GLs (GsMMQS), assuming m GLs:

(46) $$\mathbf{GsMMQS} = \frac{\sum_{k=0}^{m=k} GMMQS_k * TMS_k}{\sum_{k=0}^{m=k} TMS_k}$$

Example:

The PID GL(*k1*) has an GMMQS of 0.3 with TMS of 190 instances (see previous example). In the COPD GL (*k1*) there is 0.5 GMMQS with 60 TMS and the HypoThyrd(*k1*) 0.65 and 80 TMS instances. Thus the MQS for all GLs is:

$$\text{GsMMQS} = \frac{0.3*190 + 0.5*60 + 0.65*80}{190+60+80} = 0.42$$

Summary:

The detailed methodology for evaluation has been presented, suggesting quantitative measures of completeness, of correctness, and of completeness in various regulations: specific KR, plan, appearances of KR, group of KRs, task, and overall markup.

The ability to evaluate the markups to such a high resolution will elucidate what, and more important where, the common problems in each markup are, between EPs, between GLs, and in all the GLs. Thus, problem 8 (see section 2.4.8) has been resolved.

4.5. The Markup-Evaluation Tool

4.5.1. Starting evaluation session using MET

The evaluation methodology was implemented by developing a tool designed to produce completeness and correctness scores: The Markup-Evaluation Tool (MET) (Figure 31). The MET is a web-based desktop application which was developed using Dot.Net technology and therefore enables sharing and collaboration between different sites and users. The MET enables the EP, the KE, and other guests to select and browse the desired evaluated markup from DeGeL library. An evaluation session usually includes the relevant EP to the clinical domain of the evaluated markup, and a KE who is familiar with Asbru semantics.

There are two possible working modes in MET in each evaluation session: *View* mode and *Evaluation* mode. When one of the evaluation managers (usually the KE) starts the evaluation session, he opens MET in Evaluation mode, and enters the participants in the appropriate fields in the session: EPs, KEs and optional guests. In the View mode the users can select the markup and only view it. Thus, all other participants at different sites and locations in the session can open the MET in parallel in View mode. MET's functionality enables all participates to see the changes made by the EP and the KE who entered in Evaluation mode online during the session. When the evaluation manager enters MET, he should attach a relevant OSC file. For example, for the PID markup, the OSC of the PID GL is attached.

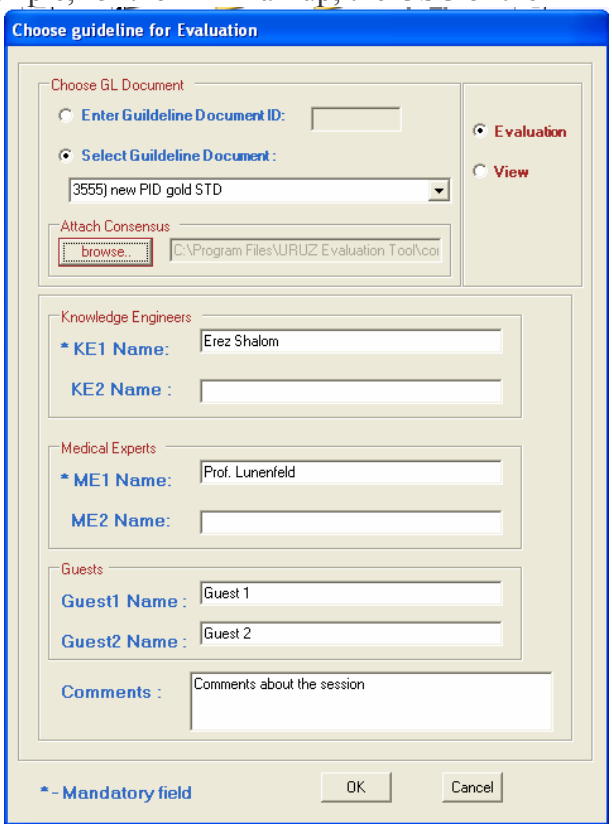


Figure 31. The Login form of The Markup Evaluation Tool. In the upper right frame the user can select the working mode (Evaluation or View). In the drop down list the user selects the appropriate markup to evaluate, and uses the "browse" button to attach an ontology specific consensus file. If the Evaluation mode is selected, the user enters the names of the KE and the EP who are going to manage the evaluation session. Note the optional fields of guests, for other casual participants

4.5.2. Evaluating the markups using MET

After the evaluation manager has filled the login form, the relevant information is downloaded from DeGeL servers and the main interface of MET is loaded. The main interface of MET is shown in Figure 32 : In the upper left panel , the tree of the plans of the markup is shown. When the evaluation manager selects a plan, its procedural content (i.e., the type of plan-body, order and plans if the type is subplan, etc.) is loaded to the upper right frame, and the declarative content (i.e., all the plans' KRs and their textual content) is loaded to the bottom right frame. The bottom left panel is used for giving the scores of completeness and correctness to the selected Krs, according to the content, and will be explained later in this section. In addition, whenever necessary, the evaluation manager, or other participants, can use multiple tabs to view the textual source of the GL, the OSC, the GL knowledge, and the Semantic Indices classification,.

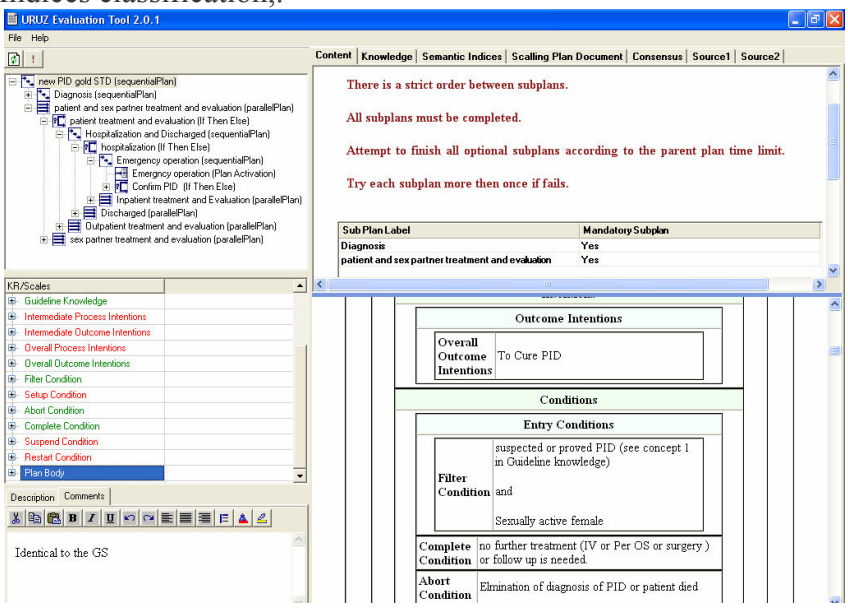


Figure32. The main interface of the MET. Note the tabs in the upper right frame. The user can switch between them whenever necessary, for example to see the textual content of the source of the particular plan in markup. Note the tree of plans in the left upper panel and the knowledge roles of the selected plan in the left bottom panel.

In the left bottom panel of the MET main interface, a list of checkboxes facilitates the scoring of the completeness and correctness measures: for each score of completeness and correctness there is a checkbox; thus, the evaluation manager, who is usually the KE, can check the desired score in the relevant checkbox during the evaluation session.

In an evaluation session, the evaluation manager together with the EP opens two instances of the MET: one for the evaluated markup and one for the GS. Then, for each plan, starting with the root plan in the GS instance, they look in the evaluated markup instance for a similar plan and decide on its completeness: they check the GS instance completeness checkboxes (Figure 33): if the plan exists in the GS and exists in the markup, or if the plan exists in the GS and is missing in the markup they check the GS instance completeness checkboxes. If the plan is missing in the GS but exists in the markup, that is, is redundant; they check the evaluated markup instance completeness checkboxes..

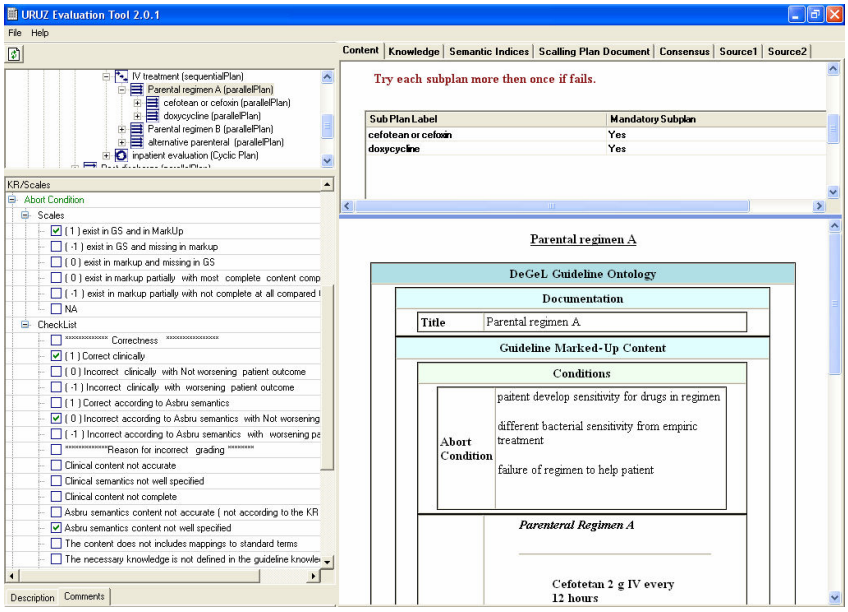


Figure33. Evaluation of the markup using MET. Note the selected plan in the left above panel. Note the checkboxes for the completeness and correctness measures in the left bottom panel, in this case of the Abort Condition knowledge role(KR). Note also the content of the procedural content of the KR in the right above panel, and its textual content in the right bottom panel

In addition, the content of each KR of each plan (existing or redundant) is evaluated according to the *Asbru* and *Clinical* correctness measures (see 4.4.1). For each score [-1,0,1] of the two correctness measures there is an appropriate checkbox which the evaluation manager can check after discuss it with the EP. Thus, during the evaluation session the KE and the EP are collaborating by checking the appropriate checkboxes of the completeness and correctness for each plan and KR in the GS and the evaluated markup instances. Additional checkboxes list each type of error (see 4.4.2), enabling them to report the error and its type, clinical or Asbru semantics error, and what kind of specific error (see 4.4.2), again by checking the appropriate checkbox (Figure 34). In addition, they might evaluate the Semantic Indices classification of the markup, by selecting its tab (Figure 34) .

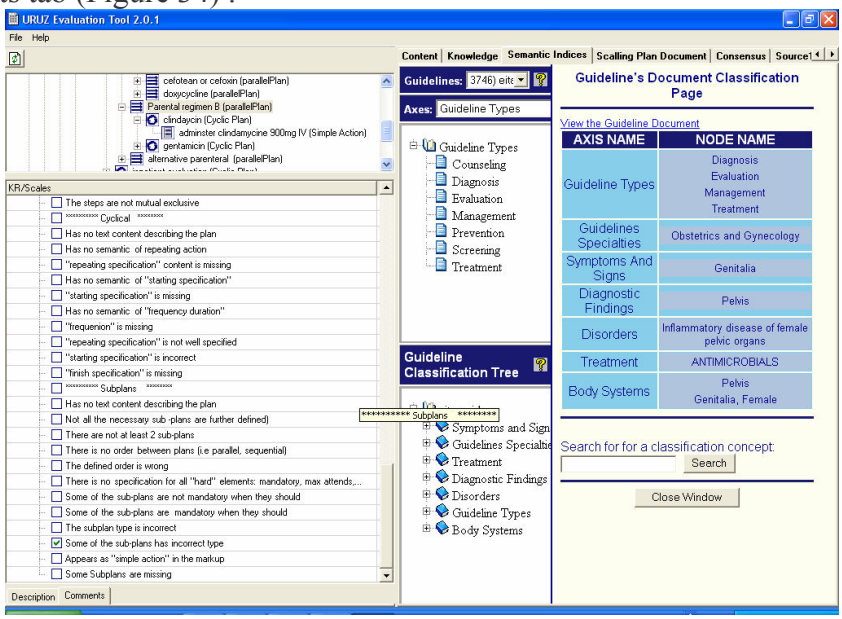


Figure34. The semantic indices view in MET. Note the checkbox list for the documentation of errors in the left bottom frame.

4.6. Research Questions and Hypothesizes

Three groups of research questions are posed in this thesis: the first group comprises questions regarding the subjective measures to analyze the questionnaires. In the second group are questions regarding the completeness objective measures, and in the third group, the objective measures of correctness in all levels of resolutions will be analyzed. The questions groups posed in this thesis are as follows:

4.6.1. Subjective Research Questions

1. (a) How did learning aspects, such as the overall process methodology and specification language, help the EP's who created an OSC for each GL? (b) Is the same aspect considered as helpful (or not) across all EPs?
   - Hypothesis:
     The EPs will give positive scores, when asked about the contribution of various aspects of the creation of the OSC. The same aspects will be helpful (or not) across all EPs.
   - Rationale for the hypothesis:
     Making an OSC requires some understanding of several aspects that are part of the overall process methodology, the specification language, and its declarative and procedural aspects, and therefore the same aspects might be considered helpful (or not) by all EPs. This hypothesis will be measured by statistical methods.
2. (a)How did learning aspects such as the overall process methodology and specification language helped the EPs who edited the GL in the markup process? (b) Is the same aspect considered as helpful (or not) across all EPs?
   - Hypothesis:
     The EPs will give positive scores when asked about the contribution of various aspects to the creation of a markup. The same aspects will be considered helpful (or not) across all EPs.
   - Rationale for the hypothesis:
     Creating a markup requires some understanding regarding several aspects such as: the overall process methodology, its tools, its specification language, and its declarative and procedural aspects. Therefore, the same aspects might be considered as helpful (or not) across all the EPs. This hypothesis will be measured by statistical methods.
3. (a) Do the aspects which seem helpful (or not) for the task of creating the OSC also seem helpful (or not) for the task of creating the markup? (b) Is (or is not) the same aspect considered as helpful across tasks by the same EPs?
   - Hypothesis:
     There is a correlation between the aspects which helped in the process of making an OSC and the aspects which helped in the markup process; there is also a correlation within the same aspects across the same EPs between the tasks
   - Rationale for the hypothesis:
     EPs use the same aspects in both phases; it might therefore be expected that what helped them in the process of making the OSC would help them in the markup process. This hypothesis will be measured by statistical methods.

4. (a) How well do EPs understand the Asbru semantics before performing markup? (b) Is (or is not) the same KR considered as easy to understand across all EPs?
   - Hypothesis:
     The EPs will give positive scores when asked about understanding of the Asbru semantics. The same KRs will be considered as easy to understand (or not) across all EPs. In particular, it will be more difficult to understand KRs which involve procedural semantics such as the *Plan-Body* KR, or more abstract ones such as *Intentions* KRs.
   - Rationale for the hypothesis:
     The preliminary step for performing the markups is to learn Asbru and its semantics; the same KRs might therefore be considered easy to understand (or not) across all the EP's. This hypothesis will be measured by statistical methods.
5. (a) How easy (or not) was it for the EPs to structure the GL according to the Asbru semantics using the URUZ Markup tool? (b) Is the same KR considered as easy to structure (or not) across all EPs?
   - Hypothesis:
     The EPs will give positive scores, when asked about the difficulty of structuring the Asbru semantics using the URUZ Markup tool. The same KRs will be considered as easy to structure (or not) across all EPs. In particular, it will be easier to structure declarative KRs and more difficult to structure procedural KRs.
   - Rationale for the hypothesis:
     The EPs were trained in the tool before marking up Asbru KRs. However, since EPs are not used to thinking in a procedural fashion, it may be assumed that it should not be an easy task with the procedural KRs. Therefore, the same KRs might be considered across all the EPs as easy to structure (or not). This hypothesis will be measured by statistical methods.
6. (a) Do the KRs of the Asbru semantics which seem easy (or not) for the EPs to understand in the phase before editing the GL also seem easy (or not) to structure in the phase of structuring the Asbru semantics using the URUZ Markup tool? (b) Is the same KR considered as easy (or not) across all EPs between these two phases?
   - Hypothesis:
     There is a correlation between the KRs which were easy to understand and easy to structure; there is also a correlation within the same KRs across the same EPs between the two phases
   - Rationale for hypothesis:
     It can be assumed that KRs which were denoted as easy (or not) to understand by the EPs were also denoted as easy (or not) to structure across all EPs. This hypothesis will be measured by statistical methods. This hypothesis will be measured by statistical methods.
7. (a) Do the KRs of the Asbru semantics which seem easy (or not) to understand and to structure as reported in the questioners by the EPs, achieved high proportion of scores of 1 in their markups? Is there correlation between the KRs reported as easy (or not) to understand and to structure and the proportion of score of 1?

- Hypothesis:
  The proportion of scores of 1 in the KRs which were reported as easy to understand and easy to structure by the EPs was high; There is a correlation between the KRs which were easy to understand and easy to structure by the EPs, and their proportion of scores of 1.
- Rationale for hypothesis:
  It can be assumed that KRs which were denoted as easy to understand and easy to structure by the EPs, will have a high proportion of score of 1 (the same is for the difficult KRs). This hypothesis will be measured by statistical methods

8. Is the URUZ markup tool usable, and to what degree?
   - Hypothesis:
     The URUZ Markup Tool will be usable.
   - Rationale for the hypothesis:
     The URUZ Markup Tool is a web-based tool supporting all the facilities of GL structuring, and the EPs are trained in its use. This hypothesis will be measured by the Scalable Usability Test (SUS) questioner (see more details about this questionnaire in section 4.3.3).

4.6.2. Questions Regarding Completeness:

1. (a)What is the level of completeness of the structuring of the marked-up GL into sub-plans? (b) Is there a significant difference in that completeness between EPs editing the same GL, and between GLs edited by the same EPs?
   - Hypothesis:
     The level of completeness as measured by the number of the plans specified within each markup out of those that were specified in the GS (in the *Existing* group – see 4.2.1) will be high, that is, considerably higher than 50%; there will be no significant difference between the two EPs editing the same GL, and between GLs edited by the same EPs.
   - Rationale for the hypothesis:
     The EPs had used an OSC; therefore, they are expected to achieve significantly high completeness scores. Since the training was similar, it is expected that these scoring will be similar across each pair of editors in a GL. This hypothesis will be measured by statistical methods.
2. (a) What is the level of completeness of the KRs and KR classes in each markup? (b) Is there a significant difference in that completeness between EPs editing the same GL, and between GLs edited by the same EPs?
   - Hypothesis:
     The level of completeness of each KR and each KR class in each markup will be high, that is, considerably higher than 50%; there will be no significant difference between the two EPs editing the same GL, and between GLs edited by the same EPs.
   - Rationale for the hypothesis:
     The EPs had used an OSC, and therefore were expected to achieve significantly high completeness scores. Since the training was similar, these scores were expected to be similar across each pair of editors in a GL, and between GLs edited by the same EPs. This hypothesis will be measured by statistical methods.

3. (a) What is the level of completeness for each GL, and for all GLs? (b) Is there a significant difference in that completeness between the GLs?
   - Hypothesis:
     The level of completeness for each GL, and for all GLs will be high, that is, considerably higher than 50%; there will be no significant difference between the GLs.
   - Rationale for the hypothesis:
     The EPs in each GL had used an OSC, and therefore they were expected to achieve significantly high completeness scores. Since the training was similar, these scores were also expected to be similar across the GLs. This hypothesis will be measured by statistical methods.
4. (a) What is the level of completeness of the KRs that compose the tasks *Application* and *QA* in each markup in a GL, and across all GLs? (b) Is there a significant difference in that level between the EPs, and between the GLs?
   - Hypothesis:
     The level of completeness of the tasks in each markup in a GL, and between all GLs will be high, that is, considerably higher than 50%; we expect no significant difference between the EPs, and between the GLs
   - Rationale for the hypothesis:
     Since all EPs used the same OSC, sources and tools, it might be assumed that there would be no significant difference in their completeness level in the different tasks. This hypothesis will be measured by statistical methods.

4.6.3. Questions Regarding Correctness

1. (a)What are the Mean Quality Scores (MQSs) of correctness for all GLs, for each of the GLs, for each markup done by an EP, for each KR class and each KR type in each GL markup? Is the proportion of scores of 1 (out of-1, 0, 1) significantly higher than 1/3 for each EP? (b) Is there a significant difference between the two EPs marking up each GL for both Asbru (syntactic) and clinical (semantic) measures, and for the overall markup? Is there a significant correlation between the two EPs marking up the same GL in those measures? Is there a significant difference in correctness between different GLs edited by the same EP in those measures?
   - Hypothesis:
     The MQS of the KR Classes and the KR types, of each markup, each of the GLs, and of all GLs will be high, that is, positive; we expect the proportion of scores of 1 to be significantly higher than 1/3 for all EPs; we expect no significant difference between the two EPs editing the same GL, and between different GLs edited by the same EP in all measures; finally, we expect a correlation between the two EPs editing the same GL in all measures.;
   - Rationale for the hypothesis:
     The EPs had used an OSC, and had some training with The URUZ tool; therefore we expect them to achieve significant high MQS scores and high proportion of scores of 1. Since the training was similar, we also expect these scoring to be similar and correlated between EPs editing the same GL, and to be similar as well between GLs edited by the same editor. This hypothesis will be measured by statistical methods.

2. (a) Is the proportion of scores of 1 (out of-1, 0, 1) is significantly higher than 1/3 for each KR class and for each KR Type? Which KR classes and KR Types were easy (or not) to structure? Is there significant difference between the KR classes and between the KR types? (b) What are the MQSs of the correctness of the Asbru and the Clinical measures in each KR class and KR type? Is there significant difference between these measures in each KR class and KR type?
   - Hypothesis:
     We expect the proportion of scores of 1 to be significantly higher than 1/3 for each KR class and for each KR Type; we expect the declarative KRs to be significantly easier to structure then the procedural KRs, that is, will have a higher mean MQS across all markups; we expect that the MQS of both measures the *Clinical* and the *Asbru* in each KR type, KR class and task will be high, that is, positive; We expect no significant difference between the *Clinical* and *Asbru* measures in each KR, and KR classes.
   - Rationale of expected result:
     The EPs had used an OSC and had some training with The URUZ tool; therefore, we expect a significant high proportion of score of 1 for each KR Class and KR type with no significant difference in both measures the Asbru and the Clinical; Since the EPs are not programmers, they less understand procedural control structures which involve logical thinking, and better understand intuitive concepts such as the declarative KRs (e.g., the *Filter Condition* KR). Therefore, we expect that the mean MQS's across all GLs will be lower for the procedural KRs. This hypothesis will be measured by statistical methods.
3. (a) What is the Mean Quality Score of the *Application* and *QA* tasks in each markup of each EP, and for each GL? Is the proportion of scores of 1 (out of-1, 0, 1) significantly higher than 1/3 for each EP in each task? (b)Is there a significant difference in that correctness between the tasks for each markup of an EP and for all markups?
   - Hypothesis:
     The Mean Quality Score of correctness of the tasks in each markup of EP will be high, that is, positive; we expect no significant difference of the tasks between the EPs, although the QA task might be easier to markup?
   - Rationale of expected result:
     The EPs had used the same OSC, and had some training with The URUZ; therefore we expect them to achieve significant high MQS scores. Since the training was similar, we also expect these scoring to be similar across the tasks. This hypothesis will be measured by statistical methods.
4. (a) What is the proportion of the general errors (see section 4.4.2) between the Clinical and Asbru measures in each GL and for all GLs? (b) What is the number of each type of error in the general errors scale for both Asbru and Clinical measures? (c) What is the number of specific errors (see section 4.4.2) of each KR for each EP, and for all EPs?
   - Hypothesis:
     We expect more errors in the Asbru measure; we expect the highest amount of general and specific errors to be in the PID GL, fewer errors in the COPD GL and smallest amount of errors in the HypoThyrd GL.
   - Rationale of expected result:
     The EPs are not programmers and thus less understand the specification language semantics. Therefore, we expect them to have more errors for the

Asbru measure than for the Clinical measure; we created the three OSCs of the three GLs with incremental level of detail, that is ,the PID was the less detailed OSC, than more detailed OSC was created for the COPD GL, and finally the most detailed OSC was created for the HypoThyrd GL;therefore, we expect that in the case of the PID GL the EPs will have the highest amount of errors, then in the COPD GL to have fewer errors, coming down to low amount of errors in the case of the HypoThyrd GL.

5. Results

5.1. Results for the Subjective Measures

5.1.1. Results regarding the aspects helped in creating an OSC

❖ Research question and its results:
(a) How did learning aspects, such as the overall process methodology and specification language, help the EPs who created an OSC for each GL? (b) Is the same aspect considered as helpful (or not) across all EPs?
(For the complete hypothesis, see hypothesis 1 in section 4.6.1)

(a) How did learning aspects, such as the overall process methodology and specification language, help the EPs who created an OSC for each GL?

- Method of measurement:
  The EPs who participated in the process of creating the OSC were given a questionnaire listing multiple aspects related to creating an OSC to score the contribution of each aspect on a scale of -3 [interfered with creating the OSC] and 3 [very helpful] (for more details, see questionnaire 1 in section 4.3.1).
- The results are shown in the following table :

Table13. The various aspects used for creating the OSC sorted by level of importance:

Aspect ID	Aspect Description	EP1	EP4	EP7	EP8	Mean	STDEV
1	Your own Medical expertise	2	3	3	3	2.75	0.50
2	Reading the guideline sources before making ont. consensus	2	2	3	3	2.50	0.58
3	Knowing the multiple representation level model	3	0	3	3	2.25	1.50
4	Asbru Krs – Procedural part	2	2	3	2	2.25	0.50
5	Asbru Krs -Declarative part	3	2	2	2	2.25	0.50
6	Ontology	1	0	2	3	1.50	1.29
7	Having more than one source	3	3	0	0	1.50	1.73
8	DeGeL	0	0	3	2	1.25	1.50
9	URUZ –main interface	1	0	0	1	0.50	0.58
10	Plan-body wizard	1	0	1	0	0.50	0.58
11	IndexiGuide	0	0	0	2	0.50	1.00
12	Vaidurya	0	0	0	1	0.25	0.50
13	Vocabulary server	0	1	0	0	0.25	0.50
14	Spock	0	0	0	0	0.00	0.00
	Mean Score	1.29	0.93	1.43	1.57	1.30	0.28

The positive mean score given by EPs, 1.3 ± 0.28 (std)(Table 13), confirms our hypothesis: knowing aspects such as the overall process methodology and the specification ontology was considered as helpful by the EPs in the process of making an OSC. In comparison with the null hypothesis of a mean score of 0, the result of a mean score of 1.3 ± 0.28 seems highly meaningful.
After sorting the various aspects by level of importance, it can be seen that the EPs' own medical expertise was listed as most helpful (2.75 ± 0.5). The next listed aspects are "reading the guideline source before making an OSC" (2.5 ± 0.58) and "knowing the hybrid model" (2.25 ± 1.51), and knowing Asbru Krs. One interesting result in particular, is that the DeGeL tools had a mean usefulness score of less than 1 point.

(b) Is the same aspect considered as helpful (or not) across all EPs?

- Method of measurement:
  Whether the same aspects were considered helpful (or not) by all the EPs was calculated by Pearson correlation test[4] between the aspect's scores for each pair of EPs
- The results are shown in the following table :

Table14. The correlation and its significance between the editors who created the OSC.

		EP1	EP4	EP7	EP8
Pearson Correlation (Sig.)	EP1	1 (.)	0.65 (0.006)*	0.47 (0.045)*	0.351 (0.109)
	EP4	0.65 (0.006)*	1 (.)	0.293 (0.155)	0.134 (0.324)
	EP7	0.47 (0.045)*	0.293 (0.155)	1 (.)	0.791 (0)*
	EP8	0.351 (0.109)	0.134 (0.324)	0.791 (0)*	1 (.)
	Mean	0.618	0.519	0.639	0.569
	Total Mean	0.586			

*: significant (P<0.05) of the correlation between the EPs

Notice that (Table 14) the correlation is rather high across EPs with total mean correlation of R=0.586 (EP1 created the OSC for the PID GL, EP4 for the COPD and EP7 and EP8 for the HypoThyrd GL). The correlation is significant between the pairs of EP1 and EP7 (R= 0.47, P=0.045), EP1 and EP4 (R= 0.65, P=0.006), EP7 and EP8 (R= 0.791, P=0) who achieved the highest significant correlation. All the correlations between the other pairs of EPs are non significant (P>0.05).

❖ Explanation for the results:
We found that aspects such as using their own medical knowledge and their understanding of Asbru semantics were considered as more helpful by the EPs than knowing the different DeGeL tools. In addition, less variability is found (which can be interpreted as showing more agreement among EPs) regarding aspects that are not helpful (such as DeGeL tools). This can be explained by the fact that the EPs involved in creating the OSC are usually senior EPs who have a lot of practical medical knowledge, but less of the computer orientation required by complex tasks (such as markup). Therefore, the DeGeL tools were considered less helpful. The significant correlation of most the EP pairs confirms this explanation. The highest correlation, between EP7 and EP8, can be explained by the fact that both of them created the same OSC of the COPD GL.

❖ Conclusion:
A preliminary future step for creating an OSC might include teaching the EPs aspects such as the specification language semantics (Asbru in our case), the concept of an ontology, the hybrid model, and having them read all of the GL sources. Teaching the different DeGeL tools is not useful.

[4] Pearson Product Moment Correlation (called Pearson's correlation for short) reflects the degree of linear relationship (correlation) between two variables. It ranges from +1 to -1. A correlation of +1 means that there is a perfect positive linear relationship between variables.

5.1.2. Results regarding the aspects helped in creating the markup

❖ Research question and its results:

(a) How did learning aspects such as the overall process methodology and specification language helped the EPs who edited the GL in the markup process?
(b) Is the same aspect considered as helpful (or not) across all EPs?
(For the complete hypothesis, see hypothesis 2 in section 4.6.1)

(a) How did learning aspects such as the overall process methodology and specification language helped the EPs who edited the GL in the markup process?

- Method of measurement:
  The EPs who participated in the process of creating a markup were given a questionnaire listing multiple aspects related to creating the markup process, and were asked to score the contribution of each aspect on a scale of -3 [interfered with the creation of the markup] and 3 [very helpful] (for more details, see questionnaire 2 in section 4.3.2)
- The results are shown in the following table:

Table15. The various aspects used for creating the markup sorted by level of importance:

Aspect ID	Aspect Description	EP1	EP2	EP5	EP8	Mean	STDEV
1	Asbru Krs – Declarative part	2	3	3	3	2.75	0.50
2	Asbru Krs – Procedural part	2	3	2	3	2.50	0.58
3	IndexiGuide	3	2	3	2	2.50	0.58
4	URUZ –main interface	3	3	2	1	2.25	0.96
5	Your Medical expertise	1	3	3	2	2.25	0.96
6	Plan-body wizard	3	3	3	-1	2.00	2.00
7	Ontology	3	0	1	3	1.75	1.50
8	Vaidurya	0	2	3	2	1.75	1.26
9	Reading the guideline sources before making ont. consensus	2	2	3	0	1.75	1.26
10	Knowing the multiple representation level model	0	0	3	2	1.25	1.50
11	Vocabulary server	1	1	3	0	1.25	1.26
12	DeGeL	0	1	1	-1	0.25	0.96
13	Spock	0	1	0	0	0.25	0.50
14	Having more than one source	-1	2	-3	0	-0.50	2.08
	Mean score	1.36	1.86	1.93	1.14	1.57	0.38

The positive mean score given by EPs, 1.57 ± 0.38 (std) (Table 15), confirms our hypothesis: knowing aspects such as the overall process methodology and the specification ontology considered as helpful to the EPs in the process of creating a markup. In comparison with the null hypothesis of a mean score of 0, the result of a mean score of 1.57 ± 0.38 seems highly meaningful.

After sorting the various aspects by level of importance, it can be seen that the aspect of knowing Asbru's declarative aspects is considered by the EPs as most helpful (2.75 ± 0.5). The next helpful aspects are Asbru's procedural aspects (2.5 ± 0.58), knowing the IndexiGuide Tool (2.5 ± 0.58), and knowing the URUZ main interface (2.25 ± 0.96). One interesting result in particular is that DeGeL framework and Spock runtime application aspects had a mean usefulness score of less then 1 point, and the aspect of having some sources even interfered with the EPs work.

(b) Is the same aspect considered as helpful (or not) across all EPs?

- Method of measurement:
  Whether the same aspects were considered helpful by all the EPs (or not) as calculated by Pearson correlation test between the aspect scores for each pair of EPs in this task
- The results are shown in the following table :

Table16. The correlation and its significance between the editors who created the markup.

		EP1	EP2	EP5	EP8
Pearson Correlation (Sig.)	EP1	1 (.)	0.337 (0.119)	0.49 (0.038)*	0.276 (0.17)
	EP2	0.337 (0.119)	1 (.)	0.196 (0.251)	0.062 (0.417)
	EP5	0.49 (0.038)*	0.196 (0.251)	1 (.)	0.278 (0.168)
	EP8	0.276 (0.17)	0.062 (0.417)	0.278 (0.168)	1 (.)
	Mean	0.526	0.399	0.491	0.404
	Total Mean	0.455			

*: significant (P<0.05) of the correlation between the EPs

Notice that (Table 16) the correlation between the EPs is not high, with a total mean correlation of R=0.455. The correlation is significant only between EP1 and EP5 (R= 0.49, P=0.038). All the correlations of other pairs of EPs are non significant (P>0.05), even between EPs who marked up the same GL.

❖ Explanation for the results:

The most helpful aspects include Asbru semantics (with the lowest variability), and, in contrast to the previous results, knowing the different DeGeL tools.

There is a low level of variability (which can be interpreted as showing more agreement among EPs) regarding the most helpful aspects, whereas having more than one source interfered at a high level of variability (which can be interpreted as showing more disagreement among EPs).

This result can be explained by the fact that performing the markup requires from the EPs to perform complex computational task. In particular, creating a markup requires knowledge of the different DeGeL tools and this might be the reason why the EPs considered the different DeGeL tools such as IndexiGuide, the URUZ interface, and the Plan Body Wizard as helpful.

The non significance correlation between most of the EP pairs can be explained by the fact that different EPs had different attitudes and opinions regarding these aspects, especially when performing complex task such as markup.

❖ Conclusion:

Before performing future markups, an EP should learn aspects such as the specification language and receive some training in use of the different DeGeL tools, especially, in our case, the IndexiGuide, URUZ interface, and Plan Body Wizard. Teaching DeGeL framework and Spock is not useful. In addition, when there is more than one source, maybe the OSC should include an indication for each plan, from which textual source it created, so that the EPs do not have to skip between the sources. However, this approach has a drawback, in that there is a tradeoff between what the OSC should and should not include. In addition, the non significant correlation between the EPs indicates that different EPs have different opinions regarding the markup process, and aspect which considered as helpful by one EP might be considered as not helpful by another.

5.1.3. Results regarding the aspects helped in both phases

❖ Research question and its results:
(a) Do the aspects which seem helpful (or not) for the task of creating the OSC also seem helpful (or not) for the task of creating the markup? (b) Is (or is not) the same aspect considered as helpful across tasks by the same EPs?
(For the complete hypothesis, see hypothesis 3 in section 4.6.1)

(a) Do the aspects which seem helpful (or not) for the task of creating the OSC also seem helpful (or not) for the task of creating the markup?

- Method of measurement:
  To test whether aspects which seem helpful (or not) for the task of creating the OSC also seem helpful (or not) for the task of creating the markup we performed Pearson Correlation test between the mean aspect scores for creating the OSC task and the mean aspect scores for creating the markup task
- The results:
  There was no correlation between the tasks (R=0.231 , P = 0.214), therefore there is no correlation between the aspects used in the OSC process and the markup.

(b) Is (or is not) the same aspect considered as helpful across tasks by the same EPs?

- Method of measurement:
  Pearson Correlation test between the aspect scores of the two tasks of the two EPs:EP1 and EP8 who participated in both tasks.
- The results are shown in the following table:

Table17. The correlation and its significance between the EPs across the tasks

	EP1	EP8
Correlations	0.111	0.554
Sig.	0.359	0.02 *

*: significant (P<0.05) of the correlation between the aspects in both tasks

The EPs who participated in the two tasks were EP1and EP8 (Table 17), and therefore we checked the correlation between them across the two tasks. There was no significant correlation for EP1 (R=0.111, P=0.359), but there was significant correlation for EP8 (R=0.554, P=0.02).

❖ Explanation for the results:
It can be seen that EP1 considered different aspects as helpful between the two tasks, whereas EP8 considered with rather significant correlation the same aspects as helpful between the two tasks. The non significant correlation between the tasks can be explained by the fact that the two tasks have different characteristics: The task of making an OSC are usually done by senior EPs, and therefore requires more theoretical aspects such as medical knowledge or specification language semantics and not understanding in componential tools. Creating a markup usually performed by EPs who have more computational orientation which requires practical understanding with DeGeL tools. Thus, different tasks require understanding and use of different aspects.

❖ Conclusion:
Although one EP considered the same aspects as helpful between the tasks, it might be concluded that creating an OSC and performing markups are different tasks; each task require teaching different aspects: the more theoretical aspects such as the EP's medical knowledge, read GL sources and knowing the specification framework will help the EPs when creating the OSC, and the other aspects which are more computerized oriented such as the specification tools (in

our case the different DeGeL tools) , will help the EPs in the markup process. The most important for both purposes, however, is to teach the EPs the specification language semantics, Asbru in our case.
In addition, maybe each task should be performed by different type of expertise: creating the OSC should be done by the seniors EPs of the medical organization, and the markup should be done by EPs who are, for example, residents and interns with less practical experience in the medical field but can perform complex computational task such as markup.

5.1.4. Results regarding understanding Asbru KRs

❖ Research question and its results:
(a) How well do EPs understand the Asbru semantics before performing markup?
(b) Is (or is not) the same KR considered as easy to understand across all EPs?
(For the complete hypothesis, see hypothesis 4 in section 4.6.1)

(a) How well do EPs understand the Asbru semantics before performing markup?

- Method of measurement:
  The EPs who participated in the process of making a markup were given a questionnaire listing all Asbru KRs and asked to score their level of understanding of each KR on a scale of -3 [difficult to understand] and 3 [very easy to understand] (for more details, see questionnaire 3 in section 4.3.2).
- The results are shown in the following table:

Table18. Asbru KRs sorted by level of understanding

KR ID	KR Description	EP1	EP2	EP5	EP8	Mean	STDEV
1	Actors	3	3	3	3	3.00	0.00
2	To be defined	3	3	3	3	3.00	0.00
3	Level of evidence	3	2	3	3	2.75	0.50
4	Strength of recommendation	3	2	3	3	2.75	0.50
5	Clinical context	3	2	3	3	2.75	0.50
6	Abort Condition	3	3	3	1	2.50	1.00
7	Complete Condition	3	3	3	1	2.50	1.00
8	Simple Action	3	2	3	2	2.50	0.58
9	Filter Condition	1	2	3	3	2.25	0.96
10	Reactivate condition	-1	1	3	3	1.50	1.91
11	Subplans – sequential order	-2	2	3	3	1.50	2.38
12	If-then -else	-3	3	3	2	1.25	2.87
13	Repeating plan	0	2	2	1	1.25	0.96
14	Subplans – any order	-3	2	3	3	1.25	2.87
15	Suspend condition	-1	2	2	1	1.00	1.41
16	Subplans – parallel order	-3	1	3	3	1.00	2.83
17	Guideline knowledge	-2	2	1	2	0.75	1.89
18	Intentions Overall - outcome	-2	3	3	-2	0.50	2.89
19	Plan activation	-3	2	3	0	0.50	2.65
20	Subplans – unorder	-3	3	-1	2	0.25	2.75
21	Intentions Overall - process	-2	2	2	-2	0.00	2.31
22	Intentions intermediate - outcome	-2	2	2	-2	0.00	2.31
23	Setup Condition	-1	1	1	-1	0.00	1.15
24	Intentions intermediate - process	-2	0	2	-2	-0.50	1.91
25	Switch- case	-3	3	-1	-1	-0.50	2.52
	Mean Score	-0.32	2.12	2.32	1.28	1.35	1.20

The positive mean score given by EPs, 1.35 ± 1.2 (std) (Table 18), confirms our hypothesis: the EPs seemed to understand Asbru semantics. In comparison with the null hypothesis of a mean score of 0, the result of a mean score of 1.35 ± 1.2 seems highly meaningful.
After sorting the various KRs by level of understanding, it can be seen that *Actors* and *To-Be-Defined* KRs are listed as most understandable (3.00 ± 0).

The next listed KRs are *Strength of recommendation, Clinical Context and Abort Condition* (2.75 ± 0.5). One interesting result is that in particular, *Setup Condition*, all the *Intentions* KR class (especially the *Intermediate Process Intention*) and *Switch Case* KR were the most difficult to understand with a mean score of less than 1. In addition, we found that declarative KRs (such as *Actors* and *Clinical Context*) received higher mean scores, with a range of scores of [1.5, 3], than procedural KRs (such as *Sub-Plans),* with a range of scores of [0.25 ,1.5].

(b) Is (or is not) the same KR considered as easy to understand across all EPs?

- Method of measurement:
  To test whether the same KRs were considered as easy to understand (or not) across all EPs we performed Pearson correlation test between the KR scores for each pair of EPs
- The results are shown in the following table:

Table19. The correlation and its significance between the editors regarding Asbru semantics before markup.

		EP1	EP2	EP5	EP8
Pearson Correlation (Sig.)	EP1	1 (.)	0.233 (0.131)	0.443 (0.013)*	0.403 (0.023)*
	EP2	0.233 (0.131)	1 (.)	-0.089 (0.337)	0.117 (0.29)
	EP5	0.443 (0.013)*	-0.089 (0.337)	1 (.)	0.366 (0.036)*
	EP8	0.403 (0.023)*	0.117 (0.29)	0.366 (0.036)*	1 (.)
	Mean	0.520	0.315	0.430	0.472
	Total Mean	0.434			

*: significant (P<0.05) of the correlation between the EPs

Notice that (Table 19) the correlation is not high between the EPs, with total mean of R=0.434. The correlation is significant only between EP1 and EP8 (R= 0.403, P=0.023), and EP5 and EP8(R= 0.366, P=0.036). All the correlations between the other pairs of EPs are non significant (P>0.05).

❖ Explanation for the results:
EPs have a better understanding of "real world" intuitive KRs such as *Actors*, *Clinical Context* and *Filter Conditions*, and therefore considered those aspects as more easy to understand. EPs have a lower level of understanding of "abstract" not intuitive KRs which are closer to the formal specification language semantics, such as *Intentions , Switch Case* and *Guideline Knowledge*. In addition, it is more difficult for EPs to understand procedural KRs with complex control structure semantics such as the *Plan-Body* KR, especially *Sub-Plan* KR and its temporal order types (such as *Unorder*, *Any order*) and therefore considered those aspects as more difficult to understand. The non significant correlation between most of the EPs can be explained by the fact that different EPs have different attitudes regarding the Asbru aspects they consider easy (or not) to understand. The significant correlation between EP5 and EP8 can be explained by the fact they used the same OSCs for the COPD and HypoThyrd GLs, and may be experienced the same level of difficulty across those GLs.

❖ Conclusion:
More emphasis needs to be made on teaching the "abstract" KRs such as *Intentions*, *Switch Case*, *and Guideline Knowledge*. In particular, more efforts are needed in teaching the procedural KRs of Asbru and their semantics. EPs using the same OSC seem to have the same opinion about aspects considered as easy to understand (or not).

5.1.5. Results regarding the difficulty of structuring Asbru KRs

❖ Research question and its results:

(a) How easy (or not) was it for the EPs to structure the GL according to the Asbru semantics using the URUZ Markup tool? (b) Is the same KR considered as easy to structure (or not) across all EPs?
(For the complete hypothesis, see hypothesis 5 in section 4.6.1)

(a) How easy (or not) was it for the EPs to structure the GL according to the Asbru semantics using the URUZ Markup tool?

- Method of measurement:
  The EPs who participated in the process of making a markup were given a questionnaire listing all Asbru KRs and asked to score the level of difficulty of structuring each KR on a scale of -3 [difficult to structure] and 3 [very easy to structure] (for more details, see questionnaire 4 in section 4.3.2)
- The results are shown in the following table:

Table20. Asbru KRs sorted by level of difficulty:

KR ID	KR Description	EP1	EP2	EP5	EP8	Mean	STDEV
1	Actors	3	3	3	3	3.00	0.00
2	To be defined	3	3	3	3	3.00	0.00
3	Clinical context	3	3	2	3	2.75	0.50
4	Simple Action	3	3	3	1	2.50	1.00
5	Complete Condition	2	3	3	1	2.25	0.96
6	Abort Condition	2	3	2	1	2.00	0.82
7	Subplans – parallel order	1	2	3	2	2.00	0.82
8	Subplans – sequential order	1	2	3	2	2.00	0.82
9	Level of evidence	0	2	3	2	1.75	1.26
10	Strength of recommendation	0	2	2	2	1.50	1.00
11	Filter Condition	2	3	-1	2	1.50	1.73
12	If-then -else	-2	3	3	1	1.25	2.36
13	Subplans – any order	-2	2	2	2	1.00	2.00
14	Suspend condition	1	2	-1	1	0.75	1.26
15	Reactivate condition	1	2	-1	1	0.75	1.26
16	Plan activation	3	2	-1	-1	0.75	2.06
17	Repeating plan	1	2	2	-2	0.75	1.89
18	Subplans – unorder	-2	2	2	1	0.75	1.89
19	Intentions Overall - outcome	-2	3	3	-2	0.50	2.89
20	Setup Condition	0	2	-1	1	0.50	1.29
21	Guideline knowledge	-2	2	3	-1	0.50	2.38
22	Switch- case	-2	2	3	-1	0.50	2.38
23	Intentions intermediate - outcome	-2	2	3	-2	0.25	2.63
24	Intentions intermediate - process	-2	2	2	-2	0.00	2.31
25	IntentionsOverall - process	-2	2	1	-2	-0.25	2.06
	Mean Score	0.32	2.36	1.84	0.64	1.29	0.97

The positive mean score given by EPs, 1.29 ± 0.97 (std) (Table 20), confirms our hypothesis: It seems that it was not difficult for the EPs to structure Asbru semantics using the URUZ markup tool. In comparison with the null hypothesis of a mean score of 0, the result of a mean score of 1.29 ± 0.97 seems highly meaningful.

After sorting the various KRs by level of difficulty, we found that *Actors* and *To-Be-Defined* KRs are listed as most understandable (3.00 ± 0). The next listed KRs are *Clinical Context* (2.75 ± 0.5) *and Simple Action* (2.5 ± 1). One interesting result is that, in particular, all the KRs of the *Intentions* KR class and the *Switch Case* KR were the most difficult to structure with a mean score of less than 1. In addition, declarative KRs (such as *Actors* and *Clinical Context*) received higher mean scores, with a ranges of scores of [2.5,3], than procedural KRs,(such as *Sub-Plans*), with a ranges of scores of [0,1].

(b) Is the same KR considered as easy to structure (or not) across all EPs?

- Method of measurement:
  To test whether the same KRs were considered as easy to structure (or not) across all the EPs we performed Pearson correlation test between the KR scores for each pair of EPs.
- The results are shown in the following table:
  Table21. The correlation and its significance between the editors regarding structuring the Asbru semantics**.**

		EP1	EP2	EP5	EP8
Pearson Correlation (Sig.)	EP1	1 (.)	0.474 (0.008)*	-0.199 (0.17)	0.554 (0.002)*
	EP2	0.474 (0.008)*	1 (.)	0.244 (0.12)	0.356 (0.04)*
	EP5	-0.199 (0.17)	0.244 (0.12)	1 (.)	0.024 (0.454)
	EP8	0.554 (0.002)*	0.356 (0.04)*	0.024 (0.454)	1 (.)
	Mean	0.457	0.519	0.267	0.484
	Total Mean	0.432			

*: significant (P<0.05) of the correlation between the EPs

Notice that (Table 21) the correlation is not high between the EPs with total mean correlation of R=0.432 (see Table 21). The correlation is significant between the pairs of EP1 and EP2 (R= 0.474, P=0.008), EP1 and EP8 (R= 0.554, P=0.002), EP2 and EP8 (R= 0.356, P=0.04). All the correlations between the other pairs of EPs are non significant (P>0.05).

❖ Explanation for the results:
EPs have a better understanding of "real world" intuitive KRs such as *Actors*, *Clinical Context* and *Filter conditions*. It was therefore easier for them to structure these KRs. However, less intuitive KRs, such as *Intentions* and *Switch Case*, are more difficult for an EP to understand and therefore to structure, as are also procedural KRs with complex semantics such as the *Plan-Body* KR, especially the *Sub-Plan* KR type. The significant correlation across most of the EPs pairs confirms this explanation. The significant correlation between EP1 and EP2 can be explained by the fact that they created markup for the same GL of PID, and therefore experienced the same difficulty of structuring in the same KRs.

❖ Conclusion:
More intuitive interfaces should be developed for the "difficult to structure" KRs such as *Intentions* and *Plan Body*, especially for the *Sub-Plan* and *Switch Case* KR types. Perhaps a graphical interface would be more appropriate. Notice that these KRs were considered by the EPs also as difficult to understand, which emphasis this conclusion.

5.1.6. Results regarding comparing understanding and structuring Asbru KRs

❖ Research question and its results:

(a) Do the KRs of the Asbru semantics which seem easy (or not) for the EPs to understand in the phase before editing the GL also seem easy (or not) to structure in the phase of structuring the Asbru semantics using the URUZ Markup tool? (b) Is the same KR considered as easy (or not) across all EPs between these two phases?

(For the complete hypothesis, see hypothesis 6 in section 4.6.1)

(a) Do the KRs of the Asbru semantics which seem easy (or not) for the EPs to understand in the phase before editing the GL also seem easy (or not) to structure in the phase of structuring the Asbru semantics using the URUZ Markup tool?

- Method of measurement:
  We performed a Pearson Correlation test between the mean KR scores for one phase and the mean aspect scores for the second phase
- The results:
  There was a high, meaningful, significant correlation between the two phases (R=0.875 , P = 0), that is, KRs of the Asbru semantics which considered as easy (or not) for the EPs to understand were also considered as easy to structure.

(b) Is the same KR considered as easy (or not) across all EPs between these two phases?

- Method of measurement:
  We performed Pearson Correlation test between the KR scores of the two phases for each EP
- The results are shown in the following table:
  The following table (Table 22) show the correlation and its significance between the two phases of understanding the Asbru semantic before the creating the markup and the difficulty of structuring it using the URUZ tool for each EP

Table22. The correlation and its significance between the EPs before and after creating markup

	EP1	EP2	EP5	EP8
Corellation	0.67	0.536	0.006	0.842
Sig.	0 *	0.003 *	0.488	0 *

*: significant correlation between the two task for the EP

Notice that all the EPs participated in the two phases, and therefore the correlation for each of them across these phases was examined (Table 22). Except for EP5, there is a significant correlation for all EPs. The correlation between groups was very high (R= 0.875) and significant (P=0); therefore, there is high correlation between the KRs considered as easy to understand by the EPs and the level of difficulty they experienced structuring them in the URUZ tool

❖ Explanation for the results:

The high correlation can be explained by the intuitive attitude of EPs: aspects they considered easy to understand were considered easy to structure. Thus, intuitive "real world" declarative KRs such as *Actors*, *Clinical Context*, and *Filter*

conditions were easy to understand and to structure, whereas less intuitive procedural KRs with complex semantic such as *Intentions* and *Switch case* and *Plan body* (especially *subplan* KR) were difficult to structure and to understand.

❖ Conclusion:
The EPs displayed consistent behavior, i.e., aspects they declared as easy to understand were also easy for them to structure. Therefore, more effort should be invested in teaching the semantics of the specification language (all Asbru KRs in our case). Perhaps a short test should be administered before markup to measure the EP's knowledge, in order to make sure the EP knows all the KRs at a good level. In addition, maybe a small simulation test of markup should be established, in order to measure EP's knowledge and understanding of the task.

5.1.7. Results regarding subjective and objective comparison

❖ Research question and its results:
(a) Do the KRs of the Asbru semantics which seem easy (or not) to understand and to structure as reported in the questioners by the EPs, achieved high proportion of scores of 1 in their markups? Is there correlation between the KRs reported as easy (or not) to understand and to structure and the proportion of score of 1? (For the complete hypothesis, see hypothesis 7 in section 4.6.1)

- Method of measurement:
  We used the results of section 5.1.4 and 5.1.5 compared to the proportion of scores of 1 for each EP (see Appendix D.2 for the all results of proportion). To test whether there is correlation between the questioners and the proportion of scores of 1 we performed Spearman Correlation Test [Siegel and Castellan 1988] [5]
- The results are shown in the following tables:
  Table 23 shows the score given by each EP in questionnaire 3 (denoted as Q.3, see section 5.1.4) scoring his understanding the Asbru KRs in scale of [-3,3], the score given by each EP in questionnaire 4 (denoted as Q.4, see section 5.1.5) scoring his difficulty of structuring each of Asbru KRs in scale of [-3,3], and the proportion of score of 1 in the different Markups of the EPs in percentage.

Table 23. The comparison between the subjective questioners and the objective measures.

		EP1			EP2			EP5				EP8			
KR Class	Krs	Q.3	Q.4	PID	Q.3	Q.4	PID	Q.3	Q.4	COPD	HypoTyrd	Q.3	Q.4	COPD	HypoTyrd
Context	Actors	3	3	100.00%	3	3	86.36%	3	3	83.33%	100.00%	3	3	8.33%	100.00%
	Clinical context	3	2	100.00%	2	3	81.82%	3	3	83.33%	100.00%	3	1	8.33%	100.00%
Intention	intermediate - outcome	-2	-2		2	2		2	3	0.00%	100.00%	-2	-1	100.00%	100.00%
	Overall - outcome	-2	-2	100.00%	3	3	100.00%	3	3	100.00%	100.00%	-2	-2	100.00%	100.00%
Condition	Abort Condition	3	2	65.00%	3	3	60.00%	3	2	60.00%	100.00%	1	1	40.00%	66.67%
	Complete Condition	3	1	100.00%	3	2	50.00%	3	3	100.00%		1	2	40.00%	
	Filter Condition	1	0	56.90%	2	2	39.74%	3	3	100.00%	100.00%	3	2	50.00%	0.00%
Plan-Body	Subplans – sequential	-2	2	85.71%	2	3	78.57%	3	-1	0.00%	66.67%	3	2	33.33%	33.33%
	If-then -else	-3	-2	96.88%	3	3	90.63%	3	3	75.00%	91.67%	2	1	37.50%	58.33%
	Cyclical plan	0	-2	78.00%	2	2	14.58%	2	2	0.00%	100.00%	1	2	75.00%	100.00%
	Subplans – parallel	-3	3	81.82%	1	2	63.64%	3	-1	12.50%	100.00%	3	-1	43.75%	75.00%
	Plan activation	-3	-2	100.00%	2	3	88.89%	3	3		100.00%	0	-2		100.00%
	To be defined	3	3	100.00%	3	3	100.00%	3	3	100.00%		3	3	100.00%	
	Simple Action	3	1	92.86%	2	2	100.00%	3	3	92.11%	92.31%	2	2	56.41%	76.92%

Notice that EP1 was rather careful with his own estimation in the questioners (that is, some of his scores were even negative), whereas in his markup he

[5] Spearman rank-order correlation measures association between two variables which requires that both variables be measured in at least an ordinal scale so that the objects or individuals under study may be ranked in two ordered series.

achieved rather high proportion. The opposite is for EP2: he was very confident about understanding and structuring the Asbru KRs (most of his scores were highly positive), but he achieved rather lower proportions in his markup. For EP5, his high confident about understanding and structuring the Asbru KRs was approved by the high proportions he achieved, especially in the HypoThyrd GL. Finally, EP8 was rather careful with his estimation of understanding and structuring the Asbru KRs, which was shown in the low proportions in his COPD markup. However in the HypoThyrd he achieved higher proportions of cores of 1.

Regarding the correlation, we found a meaningful significant correlation between the questioners scores and the proportion of scores of 1 only for EP5 and only for between his markup for the COPD GL: P=0.037 , R_s= 0.58 for the correlation between Q.3 and the markup of EP5 for the COPD, and P=0.011 , R_s= 0.68 for the correlation between Q.4 and the markup of EP5 for the COPD. All rest of pairs between the other EPs were non significantly (P>0.05).

❖ Explanation for the results:

The scores of the EPs in the questionnaire might describe their level of confidence the felt after learning the Asbru language and training in the tool: EP2 and EP5 felt highly confident, while EP1 and EP8 were rather less confident and therefore assigned rather lowers scores. The correlation of EP5 between the questionnaire, and his proportion of the COPD GL might be explained by the fact he has very high computational skills, thus when he reported that it was easy for him to understand and to structure some Asbru KR he actually achived high proportion.

❖ Conclusion:

EPs own estimation regarding understanding the specification language and the difficulty of structuring it is usually not correlated with their actual markups results. The only ones who might be correlated are EPs with good computational orientation, which are confident in their skills for performing complex tasks such as markup

5.1.8. Results regarding the usability of URUZ

❖ Research question and its results:

Is the URUZ markup tool usable?

(For the complete hypothesis, see hypothesis 8 in section 4.6.1)

- Method of measurement:

  Scalable Usability Test (SUS) [Brooke 1996] (see more details about this questionnaire in section 4.3.3).

- The results are shown in the following table:

Table 24. Scalable Usability Score test results

Question ID	Question description	EP1	EP2	EP5	EP8	STDEV
1	I think that I would like to use this system frequently	1	2	3	2	0.82
2	I found the system unnecessarily complex	4	2	2	2	1.00
3	I thought the system was easy to use	1	3	4	4	1.41
4	I think that I would need the support of a technical person to be able to use this systems	4	3	2	4	0.96
5	I found the various functions in this system were well integrated	4	2	3	4	0.96
6	I thought there was too much inconsistency in this system	2	2	2	2	0.00
7	I would imagine that most people would learn to use this system very quickly	1	5	2	1	1.89
8	I found the system very cumbersome to use	4	4	2	3	0.96
9	I felt very confident using the system	3	3	4	4	0.58
10	I needed to learn a lot of things before I could get going with this system	3	3	5	5	1.15
	SUS Score	32.5	52.5	57.5	47.5	
	Mean SUS Score	47.5				

The average SUS score was 47.5 (over 50 is usable) (Table 24). Thus, The URUZ tool is not usable

- Explanation of the results:
  Given the graphical limitations of the URUZ, the EPs cannot perform complex task such as structuring the tree of a plan using the Plan Body Wizard in an intuitive and transparent fashion. Other technical problems, such as a lack of basic operations such as "copy," "paste," and "undo," together with other web oriented problems makes URUZ cumbersome, less reliable and therefore not usable.
- Conclusion:
  A more robust, graphical, highly usable framework should be developed in order to make the structuring process easy and transparent, especially for EPs who have less computational orientation than a KE.

Summary

Creating an OSC and performing markups are different tasks; each task require different aspects to be taught: the more theoretical ones will help the EPs when making an OSC, and the other computer-oriented aspects, such as the different DeGeL tools, will help the EPs in the markup process. The most important for both purposes, however, is to teach them the specification language semantics, Asbru in this case. each task should be done by different experts: the OSC by the seniors EPs of the medical organization, and the markup by EPs who are, for example, residents and interns with less practical experience in the medical field but more able to perform complex computational task such as markup.

Regarding the markup process: the attitude of EPs regarding structuring the Asbru KR is correlated with their level of understanding. Thus, intuitive "real world" declarative KRs such as *Actors*, *Clinical Context*, and *Filter Conditions* were easy to understand and to structure, whereas less intuitive procedural KRs with complex semantics such as *Intentions* and *Switch case* and *Plan Body*, (especially *SubPlan* KR) were very difficult to structure and understand. Therefore, more effort should be invested in teaching the Asbru procedural and *Intentions* KRs. Perhaps a short test to measure the EP's knowledge should be administered before markup to make sure he knows all KRs at a good level. In addition, maybe a small simulation test of markup should be established, in order to measure the EP's knowledge and understanding of the task.

Since we can't be sure about the EPs self estimation on their performance, it is probably safer to take an editor with good computational orientation, who is confident in his skills for performing complex tasks such as markup. Perhaps medical students or resident can be appropriate.

Finally, the results showed that the URUZ markup tool is not usable. Therefore, a more robust, graphical, highly usable framework should be developed in order to make the structuring process easy and transparent.

5.2. Results for the Completeness

5.2.1. Results regarding the completeness of the markup

❖ Research question and its results:
(a)What is the level of completeness of the structuring of the marked-up GL into sub-plans? (b) Is there a significant difference in that completeness between EPs editing the same GL and between GLs edited by the same EPs?
(For the complete hypothesis, see hypothesis 1 in section 4.6.2)

(a) What is the level of completeness of the structuring of the marked-up GL into sub-plans?

- Method of measurement:
  To measure the completeness level of the plans specified in each markup formulas 2-7 (see section 4.4.4) were used
- The results are shown in the following tables:

Table 25. The number of he specified plans in each markup compared to the gold standard

PID			COPD			HypoThyrd		
GS	EP1	EP2	GS	EP8	EP5	GS	EP8	EP5
106	113	102	59	36	57	31	26	36

Notice that (Table 25) the number of plans specified for each GL was rather high: 113 and 102 plans in the markup for EP1 and EP2, respectively, in the case of PID GL; 36 and 57 for E8 and E5 in the case of COPD; and 26 and 36 for E8 and E5 in the case of the HypoThyrd GL. In most cases (except EP8), the number of specified plans was higher than in the GS.

Table 26. The completeness level in percentage by the existences groups.

		Missing group	Existing group	Redundant group	
		Exist in GS and not exist in Markup Plans(Percentage)	Exist in GS and in Markup Plans(Percentage)	Not exist in GS and exist in Markup Plans(Percentage)	Plans in GS
PID	EP1	0 (0%)	106 (100%)	1 (1%)	106
	EP2	3 (3%)	103 (97%)	0 (0%)	106
COPD	EP8	0 (0%)	59 (100%)	1 (2%)	59
	EP5	4 (7%)	55 (93%)	2 (3%)	59
HypoThyrd	EP8	3 (10%)	28 (90%)	0 (0%)	31
	EP5	2 (6%)	29 (94%)	0 (0%)	31
Mean		3%	97%	1%	65

The percentage of the plans in the *Existing* group was surprisingly high (Table 26). It is quite interesting that the number of plans is in opposite proportion to the completeness level: completeness level of 99% for 106 plans in the PID GL; 97% for 59 plans for the COPD GL, and 92% for 31 plans for the HypoThyrd, and total completeness mean of 97%. Although there is a total mean of 3% in the *Missing* group, it is quite impressive that four different editors performed such a high completeness level. Thus, the hypothesis of high completeness level of plans in each markup of an EP for each GL in the *Existing* group, is confirmed.

(b) Is there a significant difference in that completeness between EPs editing the same GL, and between GLs edited by the same EPs?

- Method of measurement:
  To test whether there is a significant difference in the completeness between the two EPs editing the same GL, we performed Proportion test [Walpole and

H.Myers 1978][6] for the number of plans specified across each pair of editors in each GL.

- The results are shown in the following tables:

Table 27. The proportion test results of the amount of specified plans between EPs in each GL. The Z value, the P value, and the confidence interval of the test is shown.

	PID	COPD	HypoThyrd
Z	1.74 *	2.03	0.47 *
P value	0.081	0.042	0.641
Confidence Interval (95%)	[-0.06 , 0.003]	[-0.132 , -0.004]	[-0.168 , 0.103]

*: non significant difference (P>0.05). note that Z values are in the acceptances interval

Table 28. The proportion test results of the amount of specified plans between the GLs of EP5 and EP8. The Z value, the P value, and the confidence interval is shown

	EP8	EP5
Z	2.43	0.06 *
P value	0.015	0.953
Confidence Interval (95%)	[-0.201 , 0.007]	[-0.111 , 0.104]

*: The non significant (P>0.05) Z values which are in the acceptances interval

Notice that (Table 27) in the case of the PID and the HypoThyrd GLs there is no significant difference in the completeness level of the plans among the EPs (p>0.05). There was also no significant difference between the GLs which EP5 structured (p>0.05) (Table 28). This result confirms our hypothesis: there is no difference in the completeness level of the plans among the EPs in each of those GLs. However, there was significant difference for the EPs in the COPD GL, and between the GLs of EP8. The results also showed that for the PID and HypoThyrd GLs and for EP5, the confidence interval around the mean difference between the proportions was very narrow. In other words, the statistical power of the test was adequate.

❖ Explanation for actual results:

The amount of the evaluated plans in the case of the PID GL was slightly smaller than the number of plans in the GS (Table 25), because it was decided when creating the GS to focus on specifying the treatment of PID, which is the main part of the GL, and not on specifying the diagnosis or sex partner treatment. EP8 always structured fewer plans than the GS, and EP5 tried to be expressive and to structure many plans. The high completeness level of the specified plans (Table 26) was probably due to the explicit OSC being a part of the markup process.

Details of the plans each editor missed can be found in Appendix D.1, but generally we found that the missing plans were usually simple actions, and not complex steps. In addition, in the same appendix there are details about the redundant plans and KRs in the markups. For example, EP2 added 35% of filter conditions.

In addition, notice that the high completeness level of the plans (Table 27) was in the same proportion among the EPs in the PID and the HypoThyrd GLs. In addition, EP5 had a constantly high completeness level in the GLs he structured (Table 28). However, in the COPD GL there is no proportion of completeness among the EPs; furthermore, there is not the same proportion of completeness in the markups of EP8 in the GLs he structured.

[6] Proportion test considers the problem of testing the hypothesis that the proportion of success in a binomial experiment of two population is equals

❖ Conclusion:
Having an expressive OSC, any EP (domain or non domain expert) can structure a GL with a large number of plans (more or less the same as the number of plans in the GS), and with a significantly high level of completeness for the same GL, and between GLs.

5.2.2. Results regarding the completeness of knowledge roles

❖ Research question and its results:
(a) What is the level of completeness of the KRs and KR classes in each markup?
(b) Is there a significant difference in that completeness between EPs editing the same GL, and between GLs edited by the same EPs?
(For the complete hypothesis, see hypothesis 2 in section 4.6.2)

(a) What is the level of completeness of the KRs and KR classes in each markup?

- Method of measurement:
  To measure the completeness level of each KR and each KR class in each markup formulas 8-16 (see section 4.4.4) were used.
- The results are shown in the following tables:

Table 29. The completeness level of an EP in each GL. The table describes the completeness level in percentages, its STD and number of marked up KRs and in KR Class in each markup of an EP in each GL.

		PID			COPD			HypoThyrd		
		EP1	EP2	num	EP8	EP5	num	EP8	EP5	num
Context	Actors	100% ± 0	82% ± 0	11	8% ± 0	83% ± 0	12	100% ± 0	100% ± 0	1
	Clinical Context	100% ± 0	80% ± 0	10	8% ± 0	83% ± 0	12	100% ± 0	100% ± 0	1
	Mean MQS Context	100% ± 0	81% ± 0.01	21	8% ± 0	83% ± 0	24	100% ± 0	100% ± 0	2
Intentions	Intermediate Outcome	N.A.	N.A.	N.A.	100% ± 0	100% ± 0	1	100% ± 0	100% ± 0	2
	Intermediate - Process	100% ± 0	75% ± 0	8	100% ± 0	100% ± 0	1	100% ± 0	100% ± 0	3
	Overall - Outcome	100% ± 0	100% ± 0	4	100% ± 0	100% ± 0	1	100% ± 0	100% ± 0	3
	Overall Process	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	100% ± 0	100% ± 0	3
	Mean MQS Intentions	100% ± 0	83% ± 0.18	12	100% ± 0	100% ± 0	3	100% ± 0	100% ± 0	11
Conditions	Filter Condition	97% ± 0	93% ± 0	28	50% ± 0	100% ± 0	2	0% ± 0	100% ± 0	1
	Abort Condition	100% ± 0	100% ± 0	9	40% ± 0	60% ± 0	5	67% ± 0	100% ± 0	3
	Suspend Condition	100% ± 0	100% ± 0	1	0% ± 0	100% ± 0	1	N.A.	N.A.	N.A.
	Reactivate Condition	100% ± 0	100% ± 0	1	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.
	Complete Condition	100% ± 0	80% ± 0	5	33% ± 0	100% ± 0	3	N.A.	N.A.	N.A.
	Set Up Condition	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	100% ± 0	100% ± 0	1
	Mean MQS Conditions	98% ± 0.01	93% ± 0.09	44	36% ± 0.22	82% ± 0.2	11	60% ± 0.51	100% ± 0	5
Plan-Body	Simple Action	100% ± 0	100% ± 0	28	100% ± 0	93% ± 0	38	85% ± 0	92% ± 0	13
	Plan Activation	100% ± 0	100% ± 0	9	N.A.	N.A.	N.A.	100% ± 0	100% ± 0	3
	If- Then -Else	100% ± 0	94% ± 0	15	100% ± 0	75% ± 0	4	100% ± 0	100% ± 0	6
	Cyclical Plan	100% ± 0	100% ± 0	24	100% ± 0	100% ± 0	2	100% ± 0	100% ± 0	2
	Subplans – Parallel Order	100% ± 0	82% ± 0	9	100% ± 0	100% ± 0	8	100% ± 0	100% ± 0	2
	Subplans – Sequential Order	100% ± 0	100% ± 0	14	100% ± 0	100% ± 0	3	67% ± 0	67% ± 0	3
	To-Be- Defined	100% ± 0	100% ± 0	4	100% ± 0	100% ± 0	4	N.A.	N.A.	N.A.
	Switch case	N.A.	N.A.	N.A.	N.A.	N.A.	N.A.	100% ± 0	100% ± 0	2
	Mean MQS Plan-Body	100% ± 0	98% ± 0.07	103	100% ± 0	94% ± 0.1	59	90% ± 0.13	94% ± 0.12	31
Total Mean MQS	Mean	100% ± 0.01	94% ± 0.09	180	70% ± 0.41	90% ± 0.12	97	90% ± 0.26	96% ± 0.08	49

The N.A. labels denotes not available data

Notice that (see Table 29) except EP8 in the case of the COPD GL (with 70% ± 0.41 of completeness), all EPs achieved more than 90% of completeness (EP1 even achieved 100% in the PID GL).
The completeness of the KR Types and KR classes was high: For the *Intentions* KR class all EPs achieved 100% of completeness. For the *Context* KR class, the EPs achieved generally high completeness, except for EP8 in the COPD GL who achieved only 8% of completeness. The *Actors* KR achieved the same level of completeness as *Clinical Context* KR.
Variability in the level of completeness with a range of [36%, 100%] was found in the *Conditions* KR class, in particular for the *Filter* and *Abort Condition* KRs, more particular, in the markups of EP8. The completeness of *Suspend and Complete Condition* KRs was high, except again for EP8 in the COPD GL.

Thus, except for EP8, all achieved a high level of completeness for the *Conditions* KR class.

The level of completeness of the *Plan-Body* KR class was very high: most of its KR types: *SubPlan-Parallel, Plan-Activation, Cyclical Plan, To-Be-define and Switch-Case* types, achieved a high level of completeness among all EPs. The *IF-Then-Else* KR type achieved a high level of completeness, except for EP5 in the COPD GL. The *Sub-Plan Sequential* also achieved 100% of completeness, except for the markups of the HypoThyrd GL. The level of completeness was also high in the *Simple Action* KR type as most of the EPs achieved more than 90% completeness.

Thus, if EP8 in the COPD GL is excluded, the high level of completeness of the KRs, KR classes, and the mean of the completeness of the EPs in each markup confirm our hypothesis: the level of completeness of KR types and classes of KRs in each markup of EP in each GL is high, that is, more than 90%.

(b) Is there a significant difference in that completeness between EPs editing the same GL, and between GLs edited by the same EPs?

- Method of measurement:

  To test whether there is a significant difference between the EPs in each GL we performed Proportion test for the completeness level across each pair of editors in each GL, and across GLs

- The results are shown in the following tables:

Table 30. The proportion test results for each GL. This table describes t he Z value, the p value and the confidence interval of the proportion test of completeness of KRs and KR classes between the EPs in each GL.

		PID			COPD			HypoThyrd		
		z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)
Context	Actors	1.48 *	0.138	[-0.41 , 0.046]	3.69	0.000	[-1.013 , -0.487]	0 *	1.000	[0 , 0]
	Clinical Context	1.48 *	0.138	[-0.41 , 0.046]	3.69	0.000	[-1.013 , -0.487]	0 *	1.000	[0 , 0]
	Mean Context	2.1	0.036	[-0.343 , -0.021]	5.21	0.000	[-0.936 , -0.564]	0 *	1.000	[0 , 0]
Intentions	Intermediate Outcome	N.A	N.A.	N.A.	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Intermediate - Process	1.49 *	0.136	[-0.448 , 0.048]	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Overall - Outcome	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Overall Process	N.A	N.A.	N.A.	N.A	N.A.	N.A.	0 *	1.000	[0 , 0]
	Mean Intentions	1.47 *	0.142	[-0.326 , 0.04]	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
Conditions	Filter Condition	0.59 *	0.553	[-0.153 , 0.082]	1.15 *	0.248	[-1.193 , 0.193]	1.41 *	0.157	[-1 , -1]
	Abort Condition	0 *	1.000	[0 , 0]	0.63 *	0.527	[-0.807 , 0.407]	1.1 *	0.273	[-0.867 , 0.2]
	Suspend Condition	0 *	1.000	[0 , 0]	1.41 *	0.157	[-1 , -1]	N.A	N.A.	N.A.
	Reactivate condition	0 *	1.000	[0 , 0]	N.A	N.A.	N.A.	N.A	N.A.	N.A.
	Complete Condition	1.05 *	0.292	[-0.551 , 0.151]	1.73 *	0.083	[-1.2 , -0.133]	N.A	N.A.	N.A.
	Set Up Condition	N.A	N.A.	N.A.	N.A	N.A.	N.A.	0 *	1.000	[0 , 0]
	Mean Conditions	1.02 *	0.306	[-0.129 , 0.04]	2.17	0.030	[-0.819 , -0.09]	1.58 *	0.114	[-0.829 , 0.029]
Plan-Body	Simple Action	0 *	1.000	[0 , 0]	1.77 *	0.077	[-0.165 , 0.007]	0.61 *	0.539	[-0.321 , 0.167]
	Plan Activation	0 *	1.000	[0 , 0]	N.A	N.A.	N.A.	0 *	1.000	[0 , 0]
	If-Then -Else	1.02 *	0.310	[-0.181 , 0.056]	1.07 *	0.285	[-0.674 , 0.174]	0 *	1.000	[0 , 0]
	Cyclical Plan	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Subplans – Parallel Order	1.48 *	0.138	[-0.41 , 0.046]	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Subplans – Sequential Order	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]	0 *	1.000	[-0.754 , 0.754]
	To-Be-Defined	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]	N.A	N.A.	N.A.
	Switch Case	N.A	N.A.	N.A.	N.A	N.A.	N.A.	0 *	1.000	[0 , 0]
	Mean Plan-Body	1.74 *	0.081	[-0.06 , 0.003]	2.03	0.042	[-0.132 , -0.004]	0.47 *	0.641	[-0.168 , 0.103]
All KRs Together		3.11	0.002	[-0.095 , -0.022]	3.4	0.001	[-0.305 , -0.087]	1.18 *	0.239	[-0.162 , 0.04]

The N.A. labels denotes not available data. The KR and KR classes with no significant difference of proportion between the EPs in the same guideline denoted with *

Notice that (Table 30) the difference of the completeness between each pair of editor marking up the same GL is usually insignificant per each individual KR. The accumulating difference in completeness overall KRs, between the two editors of each GL is much higher and in the case of the PID and the COPD GL is even significant ($p>0.05$).

Notice also that except for the completeness level of the *Context* KR class (p=0) and the mean completeness of the *Plan-Body* KR class (p=0.042) in the COPD GL, there is no significant differences in the completeness level of the KRs and KR classes among the EPs in the GLs (p>0.05).

Table 31. The proportion test results of completeness of KRs and KR classes across the COPD and the HypoTyrd GLs edited by EP8 and EP5. This table describes the Z value, the P value and the confidence interval.

		E8			E5		
		z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)
Context	Actors	2.44	0.015	[-1.073 , -0.76]	0.44 *	0.657	[-0.378 , 0.044]
	Clinical Context	2.44	0.015	[-1.073 , -0.76]	0.44 *	0.657	[-0.378 , 0.044]
	Mean Context	3.45	0.001	[-1.027 , -0.806]	0.63 *	0.530	[-0.316 , -0.018]
Intentions	Intermediate Outcome	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Intermediate - Process	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Overall - Outcome	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Overall Process	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Mean Intentions	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
Conditions	Filter Condition	0.87 *	0.386	[-1.193 , 0.193]	0 *	1.000	[0 , 0]
	Abort Condition	0.73 *	0.465	[-0.951 , 0.418]	1.26 *	0.206	[-0.829 , 0.029]
	Suspend Condition	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Reactivate condition	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Complete Condition	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Set Up Condition	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Mean Conditions	0.88 *	0.377	[-0.751 , 0.279]	1.02 *	0.308	[-0.41 , 0.046]
Plan-Body	Simple Action	2.47	0.014	[-0.35 , 0.042]	0.02 *	0.981	[-0.17 , 0.166]
	Plan Activation	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	If-Then -Else	0 *	1.000	[0 , 0]	1.29 *	0.197	[-0.674 , 0.174]
	Cyclical Plan	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Subplans – Parallel Order	0 *	1.000	[0 , 0]	0 *	1.000	[0 , 0]
	Subplans – Sequential Order	1.1 *	0.273	[-0.867 , 0.2]	-1.1 *	1.727	[-0.867 , 0.2]
	To-Be-Defined	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Switch Case	N.A.	N.A	N.A.	N.A.	N.A	N.A.
	Mean Plan-Body	2.43	0.015	[-0.201 , 0.007]	0.06 *	0.953	[-0.111 , 0.104]
All KR's Together		2.66	0.008	[-0.321 , -0.072]	1.29 *	0.196	[-0.02 , 0.14]

The N.A. labels demoted not available data; The KR and KR classes with no significant different denoted with *

Notice that (Table 31) except for the completeness level of the *Context* KR class (p=0.001) and the mean completeness of the *Plan-Body* KR (p=0.015) class of EP8, there are no significant differences in the proportion of the completeness level between his markups for the individuals KRs and for the KR classes. EP5 has the same proportion of completeness for all KRs and KR classes between his markups. These results confirm our hypothesis: there is no difference between the completeness level of KR types and KR classes among the EPs editing the same GLs, and between GLs.

- Explanation of results:

  The high difference between EP8 and EP5 in the COPD GL (Table 30) can be explained by the low level of completeness of EP8 and the high level of completeness of EP5. In addition, the low proportion of EP8 between the GLs he edited in the *Context* KR (Table 31) can be explained by the high level of completeness he achieved in the HypoThyrd GL, in contrast to the low level he achieved in the COPD GL.

- Conclusion:

  Given an Explicit OSC, any EP (domain or non domain expert) can create a markup with a high level of completeness for all KRs and KR classes. Usually,

editors for the same GLs and GLs with the same editors have no significant different in the level of completeness in their individual KRs. The completeness of *Context* and *Conditions* KR classes seems to be lower among EPs. Maybe for future teaching of EPs before markup, the importance of structuring declarative KRs should be emphasized, particularly structuring the KRs of the *Conditions* KR class , in the context that they are essential for the *Application* task, and structuring the KRs of the *Context* KR class which are essential for *QA*.

5.2.3. Results regarding the completeness of GLs

❖ Research question and its results:
(a) What is the level of completeness for each GL, and for all GLs? (b) Is there a significant difference in that completeness between the GLs?
(For the complete hypothesis, see hypothesis 3 in section 4.6.2)

(a) What is the level of completeness for each GL, and for all GLs?

- Method of measurement:
  To measure the completeness level for each GL, and all GLs, formulas 20-22 (see section 4.4.4) were used.
- The results are shown in the following table:

Table 32. The completeness level in percentages for each KR class for each GL. The number of marked up KRs in each class in each guideline and the mean completeness level for all guidelines is also shown.

	PID		COPD		HypoThyrd		Mean
	Completeness	Num	Completeness	Num	Completeness	Num	Completeness
Context	91% ± 0	21	46% ± 0	24	100% ± 0	2	68% ± 0.01
Intentions	92% ± 0.08	12	100% ± 0	3	100% ± 0	11	96% ± 0.04
Conditions	96% ± 0.05	44	59% ± 0.12	11	80% ± 0.25	5	88% ± 0.1
Plan-Body	99% ± 0.03	103	97% ± 0.05	59	92% ± 0.13	31	97% ± 0.02
Total	97% ± 0.05	180	80% ± 0.23	97	93% ± 0.15	49	91% ± 0.11

Notice (Table 32) the high level of completeness of the different GLs: the PID with the highest mean completeness of 97% ± 0.05; then, the HypoThyrd with 93% ± 0.15; and the COPD with 80% ± 0.23. The high mean completeness of all GLs, 91% ± 0.11, confirms our hypothesis: the completeness level of the GLs is high, that is, more than 90%. The *Intention* and *Plan-Body* KR classes achieved level of completeness higher than 90%, and the *Conditions* class achieved 88% ± 0.1. The *Context* class had the lowest level of completeness, because of the low completeness of the COPD GL in this class.

(b) Is there a significant difference in that completeness between the GLs?

- Method of measurement:
  To test whether there is a significant difference between the GLs, we will perform Proportion test for the completeness level across each pair of GLs
- The results are shown in the following table:

Table 33. : The proportion test results between pairs of GLs. The table shows the Z value, the p value, and the confidence interval of the proportion test of completeness between all GLs.

	PID-COPD			PID-HypoThyrd			COPD-HypoThyrd		
	z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)
Context	3.26	0.001	[-0.684 , -0.218]	0.45 *	0.656	[-0.211 , 0.029]	1.47 *	0.141	[-0.741 , -0.342]
Intentions	0.48 *	0.633	[-0.206 , 0.063]	0.9 *	0.366	[-0.206 , 0.063]	0 *	1.000	[0 , 0]
Conditions	3.69	0.000	[-0.71 , -0.11]	1.39 *	0.165	[-0.511 , 0.2]	0.97 *	0.330	[-0.712 , 0.203]
Plan-Body	1.13 *	0.260	[-0.074 , 0.025]	1.84 *	0.065	[-0.144 , 0.033]	0.67 *	0.503	[-0.129 , 0.067]
All KRs Together	4.39	0.000	[-0.242 , -0.075]	0.74 *	0.462	[-0.096 , 0.049]	2.15	0.032	[-0.238 , -0.031]

*:The KR and KR classes with no significant different

Notice that (Table 33), except for the *Context* KR class (p=0.001) and *Conditions* KR class in the PID-COPD GLs (P=0), there is no significant difference (P>0.05) in proportions of the level of completeness for all the KR classes for all GLs pairs, thus confirming the hypothesis: there is no significant difference in the level of completeness between the GLs.

❖ Explanation for the results:

The low completeness level of the *Context* and *Condition* KR classes (Table 32) can be explained by the low completeness level of the EPs in the COPD GL (see results in section 5.2.2). In addition, the declarative KR classes (*Context, and Conditions*) achieved lower completeness than the procedural KRs of the *Plan-Body* KR class. This can be explained by the fact that an explicit, very detailed OSC was used.

The non significant difference in the proportion of the completeness level for the individuals KRs between all GLs (Table 33) can be explained by the high completeness level of the EPs in all GLs, in particular, in the *Plan-Body* and *Intentions* KR classes. The difference in the completeness level in the pair of PID-COPD can be explained by the higher level of completeness of the EPs in the PID GL in both *Context* and *Condition* KR classes in contrast to the relatively lower level for the EPs in the COPD GL.

❖ Conclusion:

Given an explicit, detailed OSC, any EP (senior EP, EP, or general physician) can achieve high completeness level for individual KRs, KR classes in a GL and between GLs and for all GLs.

5.2.4. Results regarding the completeness of tasks

❖ Research question and its results:

(a) What is the level of completeness of the KRs that compose the tasks *Application* and *QA* in each markup in a GL, and across all GLs? (b) Is there a significant difference in that level between the EPs, and between the GLs?
(For the complete hypothesis, see hypothesis 4 in section 4.6.2)

(a) What is the level of completeness of the KRs that compose the tasks *Application* and *QA* in each markup in a GL, and across all GLs?

- Method of measurement:
  To measure the completeness level of the tasks in each markup in a GL and between all GLs, formulas 17-19 (see section 4.4.4) were used.
- The results are shown in the following table:

Table 34. The completeness level for the KRs that compose the tasks. This table displays the completeness level for each EP in Each GL, for each task.

	PID		COPD		HypoThyrd	
Tasks	EP1	EP2	EP8	EP5	EP8	EP5
Application	99% ± 0.01	96% ± 0.07	90% ± 0.38	92% ± 0.14	87% ± 0.32	94% ± 0.1
QA	100% ± 0	94% ± 0.1	74% ± 0.37	91% ± 0.09	93% ± 0.1	95% ± 0.09

Notice that (Table 34) the completeness of the EPs in the tasks is high: in the task of *Application*, most of the EPs achieved more than 90% of completeness, except EP8 in the HypoThyrd GL. In the *QA* task, the same trend can be noted: except from EP8 in the COPD GL who achieved 74% ± 0.37, all the EPs achieved over 90% completeness. Thus, the high completeness among the EPs in the different tasks confirms our hypothesis: the completeness level of the tasks and in each markup of EP in each GL is high, that is, more than 90%.

Table 35. The completeness level between EPs editing the same GL for each task in each GL

	PID	COPD	HypoThyrd	Mean
Task Application	98% ± 0.04	91% ± 0.21	91% ± 0.18	95% ± 0.08
Task QA	97% ± 0.05	83% ± 0.22	94% ± 0.09	92% ± 0.12

Notice (Table 35) that the level of completeness was high among the tasks between the editors of the same GL, and for all the GLs: mean completeness of 95% ± 0.08 for the *Application* task, and 92% ± 0.12 for the *QA* task. In most cases, the completeness of the *Application* task was higher than the *QA* task, in particular, for the COPD GL.

(b) Is there a significant difference in that level between the EPs, and between the GLs?

- Method of measurement:
  To test whether there is a significant difference between the EPs, and between the GLs, we will perform Proportion test for the completeness level of each task across each pair of editors in each GL, between the GLs and for all GLs.
- The results are shown in the following table:

Table 36. The proportion test results of each task between the GLs. The table shows the Z value, the p value, and the confidence interval of the proportion test of completeness of each task between all GLs.

		PID			COPD			HypoThyrd		
		z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)
Task	Application	1.91 *	0.056	[-0.067 , 0.001]	0.29 *	0.771	[-0.11 , 0.082]	1.19 *	0.233	[-0.219 , 0.052]
	QA	3.05	0.002	[-0.103 , -0.023]	2.81	0.005	[-0.274 , -0.052]	0.46 *	0.645	[-0.119 , 0.074]

*: The GLs with no significant different of the proportions

Notice that (Table 36) in the *Application* task there is same proportion of completeness for all GLs. However in the *QA* task, only the HypoThyrd has the same proportion (p>0.05).

Table 37. The proportion test results for each task between pairs of GLs. The table shows the Z value, the p value, and the confidence interval

		PID-COPD			PID-HypoThyrd			COPD-HypoThyrd		
		z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)	z statstic	P value	Confidence Interval (95%)
Task	Application	2.3	0.021	[-0.135 , 0.003]	1.94 *	0.052	[-0.156 , 0.03]	-0.04 *	1.033	[-0.109 , 0.114]
	QA	3.6	0.000	[-0.225 , -0.053]	0.31 *	0.755	[-0.079 , 0.058]	2.06	0.039	[-0.23 , -0.028]

*: The GLs with no significant different of the proportions.

Notice that (Table 37) for the *Application* task, only for the pair PID-COPD there was significant difference (p=0.021); however, for the other pairs (PID-HypoThyrd and COPD- HypoThyrd) there was non significant difference in proportions (p>0.05). For the QA task, there was non significant difference only for the PID- HypoThyrd GLs (p>0.05).

Table 38. The proportion test results between the tasks.

	z statstic	P value	Confidence Interval (95%)
Application & QA tasks	1.84 *	0.065	[-0.028 , -0.028]

Notice that (Table 38) there is no significant different in the proportions of the completeness between the tasks.

❖ Explanation for actual results:
The same proportion in the completeness level in all GLs and EPs in the *Application* task (Table 34) can be explained by the fact that it includes only the *Conditions* and *Plan-body* KR classes, which were completed at a high level by all EPs in all GLs.
For the *QA* task, the difference in proportions of the completeness between the COPD GL with the other GLs might be explained by the low completeness level of EP8 in the *Context* KR class (8%- see result 5.2.2).

• Conclusion:
Any EPs (domain or non domain expert) can create a markup with high level of completeness for the *Application* and *QA* tasks.

Summary:
Having an expressive OSC, any EP (senior EP, EP, or general physician) can structure a GL with high amount of plans (more or less as the number of plans in the GS), and with high level of completeness for KRs, KR classes, markup of a GL, and between GLs. In addition, EPs can create a markup which is complete enough for *Application* and *QA* tasks.
Finally, our conclusion is that it is not matter which EP will perform the markup- it can be domain expert, resident, intern and maybe student, the level of completeness of the markup he will specify will be always high.

5.3. Results for the Correctness

5.3.1. Results regarding the correctness of the markups

❖ Research question and its results:
(a)What are the Mean Quality Scores (MQSs) of correctness for all GLs, for each of the GLs, for each markup done by an EP, for each KR class and each KR type in each GL markup? Is the proportion of scores of 1 (out of-1, 0, 1) significantly higher than 1/3 for each EP? (b) Is there a significant difference between the two EPs marking up each GL for both Asbru (syntactic) and clinical (semantic) measures, and for the overall markup? Is there a significant correlation between the two EPs marking up the same GL in those measures? Is there a significant difference in correctness between different GLs edited by the same EP in those measures? (For the complete hypothesis, see hypothesis 1 in section 4.6.3)

(a) What are the Mean Quality Scores (MQSs) of correctness for all GLs, for each of the GLs, for each markup done by an EP, for each KR class and each KR type in each GL markup? Is the proportion of scores of 1 (out of-1, 0, 1) significantly higher than 1/3 for each EP?

- Method of measurement:
  To measure the MQS for each GL, formulas 43-47 (see section 4.4.5) were used; To measure the MQS for each KR of an EP in each GL, formulas 25-37 (see section 4.4.5) were used; To test whether the proportion of scores of 1 significantly higher than a result of 1/3 for each EP, we performed a binomial Proportions test[7] [Siegel and Castellan 1988], in which the scores of -1 and 0 were aggregated as one score versus the score of +1 for the whole GL markup of an EP.
- The results are shown in the following tables:

Table 39. The Mean Quality Score of each of the GLs, and for all GLs, categorized by KR Classes.

	PID	COPD	HypoThyrd	Mean	Mean - EP8 (COPD) Exluded
Mean MQS Context	0.84 ± 0.59	-0.08 ± 1.01	1 ± 0	0.39 ± 0.92	0.79 ± 0.61
Mean MQS Intentions	0.83 ± 0.55	0.83 ± 0.39	1 ± 0	0.89 ± 0.41	0.9 ± 0.42
Mean MQS Conditions	0.51 ± 0.58	0.18 ± 0.98	0.43 ± 0.91	0.45 ± 0.69	0.53 ± 0.63
Mean MQS Plan-Body	0.7 ± 0.64	0.56 ± 0.66	0.73 ± 0.63	0.66 ± 0.65	0.69 ± 0.65
Total Mean	0.68 ± 0.62	0.37 ± 0.83	0.77 ± 0.6	0.6 ± 0.7	0.68 ± 0.63

Notice that (Table 39) the mean MQS of all GLs was rather high (0.6 ±0.7). Notice also the relatively low MQS for the *Context* KR class in the COPD GL, but the high mean for this KR class if we exclude EP8 (COPD). In addition there is relatively lower MQS of the *Conditions* KR class in all GLs. The *Plan Body* MQS seems to be constantly higher in all GLs than the *Conditions* KR class, and the *Intentions* KR class has the highest MQS in all GLs. This result confirms our hypothesis, namely, that the MQSs for each of the GLs, and for all GLs is high, positive one.

[7] The Binomial Test procedure compares the observed frequencies of the two categories of a dichotomous variable to the frequencies expected under a binomial distribution with a specified probability parameter (1/3 in our case).

The complete tables of all the results of the correctness for each EP in each GL can be found at Appendix D.2. However, in the following table are the results for the MQSs of the markups for each EP (Table 40). Recall that the markup scores are on a scale of [-1, 0 ,1]:

Table 40. The Mean Quality Scores (MQS) of the markups for each EP. The MQS for each KR type and KR class in each markup, and its number of instances is also shown. Note the additional column for the mean, when excluding EP8 in the COPD guideline.

		PID			COPD			HypoThyrd			Mean	Mean Without EP8 (COPD)
KR Class	KRs	EP1	EP2	N	EP8	EP5	N	EP8	EP5	N		
Context	Actors	1 ± 0	0.73 ± 0.7	11	-0.83 ± 0.56	0.67 ± 0.76	12	1 ± 0	1 ± 0	1	0.4 ± 0.92	0.81 ± 0.59
	Clinical Context	1 ± 0	0.64 ± 0.82	10	-0.83 ± 0.59	0.67 ± 0.76	12	1 ± 0	1 ± 0	1	0.39 ± 0.93	0.78 ± 0.64
	Mean MQS Context	1 ± 0	0.68 ± 0.75	21	-0.83 ± 0.57	0.67 ± 0.76	24	1 ± 0	1 ± 0	2	0.39 ± 0.92	0.79 ± 0.61
Intentions	Intermediate Outcome				1 ± 0	0 ± 0	1	1 ± 0	1 ± 0	2	0.83 ± 0.39	0.8 ± 0.42
	Intermediate - Process	1 ± 0	0.55 ± 0.86	8	1 ± 0	1 ± 0	1	1 ± 0	1 ± 0	3	0.81 ± 0.55	0.83 ± 0.56
	Overall - Outcome	1 ± 0	1 ± 0	4	1 ± 0	1 ± 0	1	1 ± 0	1 ± 0	3	1 ± 0	1 ± 0
	Overall Process							1 ± 0	1 ± 0	3	1 ± 0	1 ± 0
	Mean MQS Intentions	1 ± 0	0.68 ± 0.75	12	1 ± 0	0.67 ± 0.52	3	1 ± 0	1 ± 0	11	0.89 ± 0.41	0.9 ± 0.42
Conditions	Filter Condition	0.55 ± 0.53	0.35 ± 0.58	28	0 ± 1.15	1 ± 0	2	-1 ± 0	1 ± 0	1	0.43 ± 0.61	0.45 ± 0.58
	Abort Condition	0.65 ± 0.49	0.55 ± 0.62	9	-0.2 ± 0.99	0.2 ± 1.03	5	0.33 ± 1.03	1 ± 0	3	0.46 ± 0.77	0.55 ± 0.69
	Suspend Condition	1 ± 0	1 ± 0	1	-1 ± 0	1 ± 0	1				0.5 ± 0.93	1 ± 0
	Reactivate Condition	1 ± 0	1 ± 0	1							1 ± 0	1 ± 0
	Complete Condition	1 ± 0	0.33 ± 0.78	5	-0.33 ± 1.03	1 ± 0	3				0.38 ± 0.87	0.62 ± 0.76
	Set Up Condition							1 ± 0	1 ± 0	1	1 ± 0	1 ± 0
	Mean MQS Conditions	0.65 ± 0.51	0.41 ± 0.61	44	-0.27 ± 0.97	0.64 ± 0.79	11	0.2 ± 1.03	0.67 ± 0	5	0.45 ± 0.69	0.53 ± 0.63
Plan-Body	Simple Action	0.88 ± 0.47	1 ± 0	28	0.56 ± 0.5	0.84 ± 0.54	38	0.62 ± 0.75	0.85 ± 0.66	13	0.78 ± 0.52	0.86 ± 0.5
	Plan Activation	1 ± 0	0.78 ± 0.65	9				1 ± 0	1 ± 0	3	0.92 ± 0.4	0.92 ± 0.4
	If- Then -Else	0.97 ± 0.18	0.84 ± 0.51	15	0.13 ± 0.83	0.5 ± 0.93	4	0.58 ± 0.51	1 ± 0	6	0.78 ± 0.54	0.84 ± 0.47
	Cyclical Plan	0.7 ± 0.61	-0.29 ± 0.71	24	0.75 ± 0.5	-0.5 ± 0.58	2	1 ± 0	1 ± 0	2	0.27 ± 0.83	0.24 ± 0.93
	Subplans – Parallel Order	0.73 ± 0.62	0.45 ± 0.8	9	0.31 ± 0.7	-0.06 ± 0.57	8	0.75 ± 0.5	1 ± 0	2	0.45 ± 0.72	0.47 ± 0.71
	Subplans – Sequential Order	0.75 ± 0.65	0.71 ± 0.61	14	0.17 ± 0.75	-0.67 ± 0.52	3	0 ± 0.89	0.33 ± 1.03	3	0.51 ± 0.78	0.53 ± 0.78
	To-Be- Defined	1 ± 0	1 ± 0	4	1 ± 0	1 ± 0	4				1 ± 0	1 ± 0
	Switch case							1 ± 0	1 ± 0	2	1 ± 0	1 ± 0
	Mean MQS Plan-Body	0.83 ± 0.5	0.57 ± 0.74	103	0.52 ± 0.58	0.6 ± 0.73	59	0.65 ± 0.68	0.82 ± 0.56	31	0.66 ± 0.65	0.69 ± 0.65
Total	Total Mean	0.82 ± 0.47	0.54 ± 0.71	180	0.13 ± 0.86	0.61 ± 0.73	97	0.68 ± 0.67	0.85 ± 0.52	49	0.6 ± 0.7	0.68 ± 0.63

The results of the test whether the scores of 1 significantly higher than 1/3 for each EP are in the following table (Table 41):

Table 41. The KR instances proportion of each score in each markup of an EP categorized by the scale of [-1, 0 ,1] .

	PID N=368		COPD N=194		HypoThyrd N = 96	
	EP1****	EP2**	EP8*	EP5***	EP8***	EP5****
1	85.87%	71.74%	43.30%	76.29%	79.17%	92.71%
0	10.87%	15.22%	24.74%	8.76%	10.42%	1.04%
-1	3.26%	13.04%	31.96%	14.95%	10.42%	6.25%

*: significant (P<0.05) for proportion > 0.3; **: significant (P<0.05) for proportion >0.65; ***: significant (P<0.05) for proportion > 0.7; ****: significant (P<0.05) for proportion > 0.75

The results of comparison of MQS between markups:

Notice that (Table 40) the MQS of all EPs was indeed positive and quite high (mean MQS of 0.6 ± 0.7), a result that is very encouraging. Note also, that if we exclude EP8 in the case of the COPD GL whose MQS for that GL was only 0.13 ± 0.86, the mean MQS is slightly higher (0.68 ± 0.63).

In addition, notice that the proportion of scores of 1 in all of the markups of the EPs was significantly higher then 1/3($p<0.01$) (Table 41). Furthermore, except for EP8 in the case of the COPD GL, the proportion of scores of 1 was significantly higher then 1/2 and for EP1 in the PID GL and for EP5 in the HypoThrd GL it was even significantly higher then 0.7. Thus, these results confirm our hypothesis, namely, that the proportion of scores of 1 of the EP markups is high one, positive one.

The results of comparison of mean MQSs between the KR classes and KRs:

There were not enough instances of individual marked KR for performing statistical testing for a single marked up GL thus we computed the MQSs of different KR types and KR classes as well (Table 40): In the *Intentions* KR class the MQS was highly positive (0.89± 0.41). The *Context* KR class achieved the lowest mean MQS (0.39 ± 0.92), but if we exclude the low MQS of EP8 in the case of COPD GL, the mean MQS of the *Context* KR Class is increase to 0.79 ± 0.61. All the individual KRs in the *Context* and *Intentions* classes have high MQSs for all EPs (Table 40); thus, there seems to be no clear inherent difficulty in their markup.

In the *Plan-Body* KR class the MQS's were surprisingly high (0.65 ± 0.68), but in the individual KRs there was most variability. The KRs with relatively low MQS were the less "intuitive" ones, which have more complex semantics such as *Cyclical* KR type with mean MQS of 0.27 ± 0.83, *Sub-Plan parallel* with 0.45 ± 0.72 and *Sub-Plan Sequential* with 0.51 ± 0.78. Other, more "intuitive" KR types such as *Simple-Action*, *To-Be-Defined* and *Plan Activation* seems to be understandable and easy to markup with high MQS.

Another somewhat surprising result is that marking up conditional expressions such as the *If-Then-Else* KR type did not seem to be difficult, with a MQS of 0.7 ± 0.54. If we exclude EP8 (COPD), the MQS of that KR type is raises to 0.84 ± 0.47. In addition, the conceptually difficult *Switch –Case* KR type is even more surprisingly, achieved an MQS of 1 ±0, but it only accepted in the HypoThyrd GL. (Table 40).

Finally, in the *Conditions* KR class there seems to be the highest variability of MQS's between the EPs: the score ranging from -0.27 ± 0.97 to 0.67 ± 0 with mean MQS of 0.45 ± 0.69. Note that even when we exclude the MQS of EP8

in the case of the COPD, the mean increases just slightly to (0.53 ± 0.63), which might indicate that there is some inherent difficulty in the markups, when specifying this KR class.

As to the individual KRs: we found high variability of the MQS between the EPs, especially in the KRs *Filter Condition* with mean MQS of 0.43 ± 0.61,the *Abort Condition* KR with mean MQS of 0.47 ± 0.77 and the *Complete Condition* with MQS of 0.38 ± 0.87. But, if we exclude EP8(COPD) the mean MQS of the *Complete Condition* raises to 0.62 ± 0.76, which means that the other EPs achieved a high MQS, but for the *Filter Condition* and *Abort Condition* , the mean MQS is still relatively lower, even after we exclude EP8(COPD). The Other KRs in the *Conditions* KR class such as *Reactivate* , *Suspend* and *Setup Condition* seem to have high MQS's and therefore don't seems to have a difficulty with.

(b) Is there a significant difference between the two EPs marking up each GL for both Asbru (syntactic) and clinical (semantic) measures, and for the overall markup? Is there a significant correlation between the two EPs marking up the same GL in those measures? Is there a significant difference in correctness between different GLs edited by the same EPs in those measures?

- Method of measurement:
  1. To test whether there is a significant difference between the EPs in each GL we performed the Wilcoxon Signed Ranks Test[8] [Siegel and Castellan 1988] between the two the markups for the Asbru and clinical measures, and between the overall markups. We used this test and not a standard paired t-test since the data was non-parametric, that is, we couldn't assume anything about its distribution and variance (that is, it does not have normal distribution).
  2. To test whether there is significant correlation between the EPs in each GL in the different measures, we performed a Gamma correlation test[9] [Siegel and Castellan 1988] because the data distribution was polichotomic (many instances of three distinct scores of [-1, 0, 1]), thus performing correlation test which casts the data in the form of a contingency table is more appropriate than a standard correlation test (such as Spearman test).
  3. To test whether there is significant difference in correctness between different GLs edited by the same EPs we performed Wilcoxon Mann – Whitney test[10] [Siegel and Castellan 1988] between the two GL markups of each EP. We choose this test, because the two sets are independent (that is not related) , and because the nonparametric characteristic of the data.

[8] The Wilcoxon signs ranks considered the magnitude and direction of the differences between two samples. It gives more weight to a pair which shows a large difference between the two conditions than to a pair which shows a small difference.

[9] The Gamma statistic G measures the relation between two ordinally scaled variables.

[10] The Wilcoxon Mann – Whitney nonparametric test used to test whether two independent groups have been drawn from the same population

- The results are shown in the following tables:

Table 42. The results of Wilcoxon Signed Ranks Test between the two sets of the markups, and the two aspects of Asbru and clinical measures. The Gamma correlations between the EPs in those aspects are also shown.

	PID (EP1\EP2)				COPD (EP8\EP5)				HypoThyrd (EP8\EP5)			
	Wilcoxon Test		Correlation		Wilcoxon		Correlation		Wilcoxon Test		Correlation	
	Z	P	G	P	Z	P	G	P	Z	P	G	P
Asbru part	-1.608	** **0.107**	0.527	* **0.0007**	-4.231	0.000	-0.003	0.985	-0.409	* **0.682**	0.275	0.232
Clinical part	-2.820	0.005	0.514	0.104	-2.112	0.035	-0.098	0.732	-1.556	* **0.119**	0.522	0.130
overall markup	-5.579	0.000	0.536	* **0.000**	-5.447	0.000	0.027	0.840	-1.919	* **0.054**	0.494	0.251

*: significant (P<0.05) for the Gamma correlations between the EPs; **: significant (P<0.05) for showing there is no difference between the scores of the EPs using Wilcoxon test

Notice that (Table 42) only for the HypoThyrd GL there is no significant difference between the two EPs (P>0.05) in all measures. Surprisingly, in the PID GL, there is no significance different between the EPs in the Asbru measure. In this GL too (the PID GL) there is the only significant (but not highly meaningful) correlation between the EPs in their Asbru measure and in their overall markups. Surprisingly also, there in this GL too there is a significance difference (P>0.05) of the clinical measure among the EPs. In the However, in the case of the COPD GL there was significant different between the markups of the EPs (P<0.05) which were also not correlated.

Regarding the results of EP editing different GLs, we performed the test on the markups of EP5 and EP8 between the COPD and the HypoThyrd GLs. However, we found significant difference (P<0.02)) between their markups for all measures (Asbru, Clinical and overall markup.

❖ Explanation for all the results:

In Table 39 the low MQS in the *Context* KR class in the COPD and the Mean of all GLS in the *Context* KR class can be explained by the low mean of EP8 (-0.83 ± 0.57). In Table 40 and 41 the significant high proportion (more than 1/3) of all the EPs can be explained by the fact they had some training before markup, and used an explicit OSC. The relatively low MQS in the markup of EP2 for the PID GL (0.54 ±0.71) can be explained by the low MQS he achieved in the *Condition* KR class, in particular for the *Filter* and *Complete Condition* KRs, probably because of some semantics errors in the markup. In the *Plan-Body* KR class, EP2 had particular difficulty with the *Cyclical* KR type with an MQS of -0.29 ± 0.71, probably resulting from an insufficient understanding of the complex semantic of this KR.

The lower MQS of EP8 in the COPD (0.13 ± 0.86) can be explained by his very low MQS in the *Context* KR class of -0.83 ± 0.57, probably resulting from his low completeness level (only 8%). In addition, he achieved a low MQS of -0.27 ± 0.97 in the *Conditions* KR class; in particular he had a problem with the *Abort Condition* and *Complete Condition* KRs, again because of his low completeness levels(see result 5.2.2), although his MQS improved between his two markups.

The variability in the MSQ of the *Conditions* KR class can be explained by the low MQS of EP2 in the PID GL and of EP8 in both COPD and HypoThyrd GLs. The difficulties in structuring the *Filter*, *Abort*, and *Complete Conditions* KRs can be explained by the fact they were extensively used in all GLs (in contrast to the *Suspend*, *Reactivate* and *Setup* which are partially used and have a higher MQS), and thus the probability of marking it up incorrectly is increased. In addition, in the PID GL the OSC deliberately excluded "AND/Or" operators in the conditions (see section 4.2.3 for explanation), increasing the amount of errors in the *Conditions* KR class, and in its composing KRs.

The relatively low MQS in the *Plan-Body* KR class in the types of *Cyclical* and *Sub-Plans* with it both parallel and sequential types, can be explained by the fact that these KRs are less "intuitive" for EPs because they include complex abstract semantics of procedural structure that is hard for EPs to understand, and therefore hard to markup with the URUZ, especially when it is neither usable and nor user-friendly (see result 5.1.7).

We found that EPs understand simple semantic "real world" KRs better, such as *Simple Action*, *Plan-activation* and *To-Be-defined* KRs (see results 5.1.4), therefore the MQS they achieved for these KRs are high. The surprising result of high MQS in the conditional type *If-Then-Else*, can be explained by the fact that this is an intuitive condition that is easy to understand, probably because the EPs are used to thinking in this semantics in their daily decisions, thus within the process of making an OSC, conditional branches were easily drawn by the EPs (see section 4.1.4);. The high MQS of the *Switch-Case* KR type surprisingly can be explained by the fact it was marked up only in the HypoThyrd GL by two EPs, who both achieved a high MQS for this type as well.

In Table 42, the result of nonsignificant difference between the EPs in the HypoThyrd GL can be explained by the fact that they used a detailed, structured OSC, which was the most detailed OSC among the three OSCs (see section 4.2.3 for explanation), and included explicitly for example the denotation if some sub-plan is mandatory or not . The difference between the EPs in the PID GL in the Clinical measure and not in the Asbru measure, can be explained by the fact that in spite of the fact that they used an OSC, which helped them to structure the GL according to Asbru semantics, they still had a different interpretation regarding the clinical aspects of the GL. Another explanation might be that one of the EP didn't follow the OSC clinical definitions. The correlation in the Asbru measure and the overall markup between the EPs in the case of the PID GL might be explained by the fact that both EPs were Senior EPs, and therefore experienced the same difficulties in the specification process for the same KRs. Finally, the significant difference between different GLs edited by the same EP can be explained by the different level of the complexity of the GLs, the different level of detail of the OSC the EPs used and the different level of experience with the URUZ tool the EPs had between the GLs, i.e. as the EPs were more experience with the tool between the two markups their MQS increased, (in the case of EP8 it increased in 0.5 in the scale of [-1,1]).

- Conclusion :

With some training, and an explicit OSC, EPs can perform markups with high correctness for each GL and between GLs, with a significant high proportion of scores of 1. In addition, in order to increase the quality of the markup of the KRs for the *Condition*s and *Plan-Body* KR classes, a more intuitive, graphic, user friendly interfaces should be used in order to bring to minimum the need from the EP to understand the formal Asbru semantics and its complex KRs (such as *Cyclical*, *Sub-Plans* or *Abort Condition*), and to bridge the gap between the initial structuring of the EP and the semantics of the specification language.

In addition, we should use a structured OSC as detail as possible, to decrease the variability between the EPs marking up the same GL. Although we saw that the OSC helped to structure the Asbru semantics, there still be disagreement and different interpretations of the GL, even when using an OSC, a fact that further emphasizes the need for a more detailed OSC. It might also be conjectured that a less experienced EP could better follow the OSC. The comprehension of the target ontology might be more important.

5.3.2. Results regarding the correctness of the KRs

❖ Research question and its results:
(a) Is the proportion of scores of 1 (out of-1, 0, 1) is significantly higher than 1/3 for each KR class and for each KR Type? Which KR classes and KR Types were easy (or not) to structure? Is there significant difference between the KR classes and between the KR types? (b) What are the MQSs of the correctness of the Asbru and the Clinical measures in each KR class and KR type? Is there significant difference between these measures in each KR class and KR type?
(For the complete hypothesis, see hypothesis 2 in section 4.6.3)

(a) Is the proportion of scores of 1 (out of-1, 0, 1) is significantly higher than 1/3 for each KR class and for each KR Type? Which KR classes and KR Types were easy (or not) to structure? Is there significant difference between the KR classes and between the KR types?

- Method of measurement:
  1. To test whether the proportion of scores of 1 significantly higher than 1/3 for each KR class and for each KR Type, we performed a binomial proportions test, in which the scores of -1 and 0 were aggregated as one score versus the score of +1 for the whole KR class and Kr Type.
  2. To test which are the easy (or not) KR classes to structure we sorted the KR classes by their MQS and then by their proportion of scores of 1. To test whether there is significant difference between them, we performed Proportion test for the number of scores of 1 between each pair of classes (the same proportion test we performed for the completeness, see 5.2.2). In addition, we performed the Kruskal-Wallis test[11] [Siegel and Castellan 1988] (again, because of the nonparametric characteristic of the data) for finding homogenous groups of KR classes. We used the Man-Whitney test for independent pairs between each pair of classes (as we performed in the previous hypothesis, see 5.3.1) in order to test if there is significant different between each pair of classes.
  3. To test whether the proportion of scores of 1 significantly higher than 1/3 for each KR Type, we performed a binomial proportions test, in which the scores of -1 and 0 were aggregated as one score versus the score of +1 for the whole KR Type. To test what are the easy (or not) KRs to structure we sorted the KRs by their MQS and by their proportion of scores of 1. To test whether there is significant difference between them, we performed Kruskal-Wallis test [Siegel and Castellan 1988] for finding homogenous groups of KR Types, and then performed proportion test of scores of 1 between each pair of homogenous group we found, to test if the difference between them is significant

[11] The Kruskal-Wallis one-way analysis of variance by ranks is used for testing whether k independent samples come from the same population or from identical populations with the same median

Before showing the result, it should be noted that we excluded in each KR class the KRs with not enough instances (less than 12) for statistical test. Those KRs are shown in the following table (Table 43):

Table 43. The excluded KRs

KR Type	Exluded from KR Class	Num of instances
Overall Process Intentions	Intentions	12
Intermediate Outcome	Intentions	12
Restart Condition	Conditions	4
Setup Condition	Conditions	4
Suspend Condition	Conditions	8
Switch case	Plan-Body	2

- The results are shown in the following tables:

Results for KR Classes:

The sorted classes according to their MQS, divided by the average score are shown in table 44:

Table 44. The ranked classes according to their MQS

Rank Status	Place	KR Classes	Mean
More than Average	1	Intentions	0.89 ± 0.41
	2	Plan-Body	0.66 ± 0.65
Less than Average	3	Conditions	0.45 ± 0.69
	4	Context	0.39 ± 0.92
		All KR Classes	0.6 ± 0.7

The result of sorting the KR classes by the proportion of scores of 1 in each KR class, and the significant for proportion greater than 1/3 its are shown in table 45:

Table 45. KR classes sorted to the left by the proportion of scores of 1 .the raw statistic (Z value) is also shown in table

Propotions of scores of 1	93.02%	75.87%	69.68%	38.10%
KR Class / P Value	**Intentions**	**plan- body**	**Context**	**Conditions**
P>0.3	0.42 *	0.36 *	0.39 *	0.38
P>0.5	0.59 *	0.53 *	0.56 *	0.55
P>0.6	0.69 *	0.63 *	0.66 *	0.65
P>0.65	0.73 *	0.68 *	0.71	0.70
P>0.7	0.78 *	0.73 *	0.75	0.75
P>0.75	0.83 *	0.78	0.80	0.79

*: significant (P<0.05) for proportion greater than the one in the "P value "cell

Notice that (Table 45) when sorting the KR classes by their MQS the *Condition* KR class is third, and when sorting the KR classes by the proportion of scores of 1 it fourth and last, and don't have a significant proportion of scores of 1 higher even than 1/3.

Testing the difference between the KR classes

For testing whether there is significant difference between the KR classes we used Kruskal-Wallis test for finding homogenous groups of KR classes and found that at least one KR classes is significantly different form the others. Then, we performed the Man-Whitney test between each pair of KR classes and found a significant difference (P<0.05) between all KR classes pairs.

Finally, for refining the testing we performed proportions test of score of 1 between each pair of KR classes, and found a significant difference in all KR classes pairs ($P<0.05$), except for one pair of KR classes, namely, the *Context* and *Plan-Body* KR classes pair which its difference was non-significant ($Z=1.45$, $P=0.08$).

Results for KR Types:

The sorted KRs according to their MQS , divided by the average score are shown in table 46:

Table 46. The ranked KRs according to their MQS

Rank Status	Place	KRs	Mean MQS
More than Average		To be Defined	1 ± 0
	2	Plan Activation	0.92 ± 0.4
	3	Intermediate Outcome Intentions	0.83 ± 0.39
	4	Intermediate Process Intentions	0.81 ± 0.55
	5	Simple Plan	0.78 ± 0.52
	6	If Then Else	0.78 ± 0.54
More than Average	7	Sequential Plan	0.51 ± 0.78
	9	Abort Condition	0.46 ± 0.77
	10	Parallel Plan	0.45 ± 0.72
	11	Filter Condition	0.43 ± 0.61
	12	Actors	0.4 ± 0.92
	13	Clinical Context	0.38 ± 0.93
	14	Complete Condition	0.38 ± 0.87
	15	Cyclic	0.27 ± 0.83
		All KRs	0.6 ± 0.7

The result of sorting the KRs by their proportion of scores of 1 in each KR, and the significant for proportion greater than 1/3 are shown in table 47 in the next page.

Note that TBD KR type has the highest proportions of scores of 1 (100%), and Surprisingly the *Filter Condition* KR has the lowest proportion (49.33%).All other KRs are raging between them. For more detail results of proportions of the different KRS see Appendix D.3

In all KRs, the proportions of score of 1 is significantly ($P<0.05$) higher than 1/3. In some KRs the proportion was significantly higher than 0.75!.

After sorting the KRs, we recognized three main groups of KRs:

1) **Easy** to structure KRs which their proportion of scores of 1 was significantly higher than 0.75;

2) **Medium** difficulty to structure, which their proportion of scores of 1 was significantly higher than 0..6 and 0.5;

3) **Difficult** to structure, which their proportion of scores of 1 was significantly higher only than 0.3.

When we performed Kruskal-Wallis test on each set of KRs in each group, we found that there was no significant difference ($P>0.05$) between the KRs in each group, thus each group is homogenous. Finally, we performed proportions test on the scores of 1 in each pair of group and found that in all pair of groups the proportion of scores of 1 is significantly different ($P=0$)

The KRs, their KR class and their group are listed in table 48 in the next page..

Table 47. KR sorted to the left by the proportion of scores of 1 .The number of instances of scores 1 in each KR and the raw statistic (Z value for each P-value) is also shown

Propotions	100.00%	100.00%	95.83%	88.89%	83.65%	83.49%	69.79%	69.57%	68.35%	62.86%	62.50%	57.65%	50.89%	49.33%
N	32	32	48	54	104	315	96	92	79	70	32	85	112	150
KR / P Value	TBD	Overall Outcome Intentions	Plan Activation	Intermediate Process Intentions	IfThenElse	simple Action	Actors	Clinical Context	sequential Plan	Abort Condition	Complete Condition	parallel Plan	cyclic	FIlter Condition
P>0.3	0.47 ****	0.47 ****	0.44 ****	0.44 ****	0.41 ****	0.38 ****	0.41 ***	0.41 ***	0.42 **	0.43 **	0.47 *	0.42 *	0.41 *	0.4 *
P>0.5	0.64 ****	0.64 ****	0.62 ****	0.61 ****	0.58 ****	0.55 ****	0.58 ***	0.59 ***	0.59 **	0.6 **	0.64	0.59	0.58	0.57
P>0.6	0.74 ****	0.74 ****	0.72 ****	0.71 ****	0.68 ****	0.65 ****	0.68 ***	0.68 ***	0.69	0.7	0.74	0.69	0.68	0.67
P>0.65	0.79 ****	0.79 ****	0.76 ****	0.76 ****	0.73 ****	0.69 ****	0.73	0.73	0.74	0.74	0.79	0.73	0.72	0.71
P>0.7	0.83 ****	0.83 ****	0.81 ****	0.8 ****	0.77 ****	0.74 ****	0.78	0.78	0.78	0.79	0.83	0.78	0.77	0.76
P>0.75	0.88 ****	0.88 ****	0.85 ****	0.85 ****	0.82 ****	0.79 ****	0.82	0.82	0.83	0.83	0.88	0.83	0.82	0.81

* denotes significant (P<0.05) for proportion greater than the one in the "proportions" cell: **** for P value >0.75; *** for P value >0.6; *** for P value >0.5; * for P value >0.3.

Table 48. The different groups of difficulty, the KRs in each group and its KR class

Group	KR	KR Class
Easy	TBD	Plan-Body
	Overall Outcome Intentions	Intention
	Plan Activation	Plan-Body
	Intermediate Process Intentions	Intention
	IfThenElse	Plan-Body
	simple Action	Plan-Body
Medium	Actors	Context
	Clinical Context	Context
	sequential Plan	Plan-Body
	Abort Condition	Conditions
Difficult	Complete Condition	Conditions
	parallel Plan	Plan-Body
	cyclic	Plan-Body
	FIlter Condition	Conditions

(b) What are the MQSs of the correctness of the Asbru and the Clinical measures in each KR class and KR type? Is there significant difference between these measures in each KR class and KR type?

- Method of measurement:
  To measure the MQS for the correctness measure of Clinical and Asbru, formulas 25-32 (see section 4.4.5) were used. To test whether there is significant difference between the Asbru and Clinical measures, we performed Proportion test for the number of scores of 1 between the Asbru and the Clinical measures in each KR class and KR type
- The results are shown in the following tables:

  The Asbru and Clinical MQS in each KR Class
  The MQSs of both the Clinical and the Asbru correctness measures in each KR class are shown in the following table (Table 49). More details tables can be found in appendix D.2.

Table 49. The MQSs of both measures of correctness: clinical and Asbru in each KR class

		PID		COPD		HypoThyd		All
		EP1	EP2	EP8	EP5	EP8	EP5	Mean
Context	Clinical	1 ± 0	0.71 ± 0.72	-0.83 ± 0.58	0.67 ± 0.76	1 ± 0	1 ± 0	0.4 ± 0.92
	Asbru	1 ± 0	0.62 ± 0.8	-0.83 ± 0.58	0.67 ± 0.76	1 ± 0	1 ± 0	0.38 ± 0.93
	Mean	1 ± 0	0.68 ± 0.75	-0.83 ± 0.57	0.67 ± 0.76	1 ± 0	1 ± 0	0.39 ± 0.92
Intentions	Clinical	1 ± 0	0.62 ± 0.77	1 ± 0	0.67 ± 0.58	1 ± 0	1 ± 0	0.89 ± 0.42
	Asbru	1 ± 0	0.62 ± 0.77	1 ± 0	0.67 ± 0.58	1 ± 0	1 ± 0	0.89 ± 0.42
	Mean	1 ± 0	0.68 ± 0.75	1 ± 0	0.67 ± 0.52	1 ± 0	1 ± 0	0.89 ± 0.41
Conditions	Clinical	0.93 ± 0.33	0.63 ± 0.53	-0.27 ± 1.01	0.64 ± 0.81	0.2 ± 1.1	0.67 ± 0.82	0.64 ± 0.7
	Asbru	0.33 ± 0.48	0.18 ± 0.51	-0.18 ± 0.98	0.64 ± 0.81	0.2 ± 1.1	0.67 ± 0.82	0.26 ± 0.64
	Mean	0.65 ± 0.51	0.41 ± 0.61	-0.27 ± 0.97	0.64 ± 0.79	0.2 ± 1.03	0.67 ± 0	0.45 ± 0.69
Plan Body	Clinical	0.81 ± 0.57	0.52 ± 0.83	0.88 ± 0.42	0.54 ± 0.79	0.74 ± 0.68	0.81 ± 0.6	0.7 ± 0.69
	Asbru	0.87 ± 0.41	0.62 ± 0.54	0.15 ± 0.48	0.63 ± 0.67	0.52 ± 0.68	0.83 ± 0.53	0.63 ± 0.6
	Mean	0.83 ± 0.5	0.57 ± 0.74	0.52 ± 0.58	0.6 ± 0.73	0.65 ± 0.68	0.82 ± 0.56	0.66 ± 0.65
ALL	Clinical	0.88 ± 0.46	0.58 ± 0.76	0.34 ± 0.93	0.59 ± 0.77	0.76 ± 0.66	0.84 ± 0.55	0.66 ± 0.72
	Asbru	0.77 ± 0.46	0.49 ± 0.65	-0.09 ± 0.73	0.64 ± 0.7	0.61 ± 0.67	0.86 ± 0.5	0.54 ± 0.68
	Mean	0.82 ± 0.47	0.54 ± 0.71	0.13 ± 0.86	0.61 ± 0.73	0.68 ± 0.67	0.85 ± 0.52	0.6 ± 0.7

Notice that in almost all KR classes both measures are quite high. The KR classes with relatively lower MQS in both measures are the *Context* KR class (with mean MQS of 0.4 ± 0.92 for the *Clinical* measure and mean MQS of 0.38 ± 0.93 for *Asbru* measure), and the *Conditions* KR class, that its *Asbru* measure is lower than its *Clinical* measure.

In most of the KR classes among the markups, the *Clinical* measure is equal or greater then the *Asbru* measure, but surprisingly, in the *Plan-Body* KR class and, except the markups of EP8, the MQS of the *Asbru* measure was higher then the *Clinical* measure. In fact, for EP5, the *Asbru* measure is equal or greater the *Clinical* measure in all KR classes in both his PID and HypoThyrd GL markups.

When we performed the proportion test of the number of scores of 1 in the set of Asbru and the set of Clinical in each KR class, we found that there was non significant difference (P>0.05) in the proportions in *Intentions* and *Context KR* classes. However, in all other classes, a significant difference between the Asbru and the Clinical measure was found (See appendix D.4 for the complete results).

The Asbru and Clinical MQS in each KR Type

The MQSs of both, the clinical and the Asbru correctness measures in each KR type are shown in the following tables (Tables 50 and 51). More details tables can be found in appendix D.2.

Declarative KRs:

Table 50. The MQS of both measures of correctness: clinical and Asbru in each declarative KR type . The number of instances is also shown

			PID			COPD			HypoThyrd			
			EP1	EP2	N	EP8	EP5	N	EP8	EP5	N	Mean
Context	Actors	clinical	1 ± 0	0.82 ± 0.6	11	-0.83 ± 0.58	0.67 ± 0.78	12	1 ± 0		1	0.42 ± 0.92
		Asbru	1 ± 0	0.64 ± 0.81		-0.83 ± 0.58	0.67 ± 0.78		1 ± 0			0.38 ± 0.94
		Mean MQS	1 ± 0	0.73 ± 0.7		-0.83 ± 0.56	0.67 ± 0.76		1 ± 0			0.4 ± 0.92
	Clinical Context	clinical	1 ± 0	0.64 ± 0.84	11	-0.83 ± 0.6	0.67 ± 0.78	12	1 ± 0		1	0.38 ± 0.93
		Asbru	1 ± 0	0.64 ± 0.84		-0.83 ± 0.6	0.67 ± 0.78		1 ± 0			0.38 ± 0.93
		Mean MQS	1 ± 0	0.64 ± 0.82		-0.83 ± 0.59	0.67 ± 0.76		1 ± 0			0.38 ± 0.93
Intentions	Intermediate Process Intentions	clinical	1 ± 0	0.55 ± 0.88	10	1 ± 0	1 ± 0	1	1 ± 0		3	0.81 ± 0.56
		Asbru	1 ± 0	0.55 ± 0.88		1 ± 0	1 ± 0		1 ± 0			0.81 ± 0.56
		Mean MQS	1 ± 0	0.55 ± 0.86		1 ± 0	1 ± 0		1 ± 0			0.81 ± 0.55
	Intermediate Outcome Intentions	clinical				1 ± 0	0 ± 0	1	1 ± 0		2	0.83 ± 0.41
		Asbru				1 ± 0	0 ± 0		1 ± 0			0.83 ± 0.41
		Mean MQS				1 ± 0	0 ± 0		1 ± 0			0.83 ± 0.39
	Overall Process Intentions	clinical							1 ± 0		3	1 ± 0
		Asbru							1 ± 0			1 ± 0
		Mean MQS							1 ± 0			1 ± 0
	Overall Outcome Intentions	clinical	1 ± 0	1 ± 0	4	1 ± 0	1 ± 0	1	1 ± 0		3	1 ± 0
		Asbru	1 ± 0	1 ± 0		1 ± 0	1 ± 0		1 ± 0			1 ± 0
		Mean MQS	1 ± 0	1 ± 0		1 ± 0	1 ± 0		1 ± 0			1 ± 0
Conditions	Filter Condition	clinical	0.9 ± 0.16	0.62 ± 0.59	29	0 ± 1.41	1 ± 0	2	-1 ± 0		1	0.7 ± 0.59
		Asbru	0.21 ± 0.43	0.08 ± 0.42		0 ± 1.14	1 ± 0		-1 ± 0			0.15 ± 0.49
		Mean MQS	0.55 ± 0.53	0.35 ± 0.58		0 ± 1.15	1 ± 0		-1 ± 0			0.43 ± 0.61
	Abort Condition	clinical	1 ± 0	0.8 ± 0.67	10	-0.2 ± 1.1	0.2 ± 1.1	5	0.33 ± 1.15		3	0.67 ± 0.81
		Asbru	0.3 ± 0.48	0.3 ± 0.5		-0.2 ± 1	0.2 ± 1.1		0.33 ± 1.15			0.33 ± 0.72
		Mean MQS	0.65 ± 0.49	0.55 ± 0.62		-0.2 ± 0.99	0.2 ± 1.03		0.33 ± 1.03			0.46 ± 0.77
	Complete Condition	clinical	1 ± 0	0.33 ± 0.82	5	-0.33 ± 1.15	1 ± 0	3				0.38 ± 0.89
		Asbru	1 ± 0	0.33 ± 0.82		-0.33 ± 1.15	1 ± 0					0.38 ± 0.89
		Mean MQS	1 ± 0	0.33 ± 0.78		-0.33 ± 1.03	1 ± 0					0.38 ± 0.87
	Suspend Condition	clinical	1 ± 0	1 ± 0	1	-1 ± 0	1 ± 0	1				0.5 ± 1
		Asbru	1 ± 0	1 ± 0		-1 ± 0	1 ± 0					0.5 ± 1
		Mean MQS	1 ± 0	1 ± 0		-1 ± 0	1 ± 0					0.5 ± 0.93
	Restart Condition	clinical	1 ± 0	1 ± 0	1							1 ± 0
		Asbru	1 ± 0	1 ± 0								1 ± 0
		Mean MQS	1 ± 0	1 ± 0								1 ± 0
	Setup Condition	clinical							1 ± 0		1	1 ± 0
		Asbru							1 ± 0			1 ± 0
		Mean MQS							1 ± 0			1 ± 0

Procedural KRs:

Table 51 The MQS of both measures of correctness: clinical and Asbru in each procedural KR type

		PID		COPD		HypoThyrd		ALL
		EP1	EP2	EP8	EP5	EP8	EP5	
Cyclic	Clinical	0.64 ± 0.7	-0.46 ± 0.83	1 ± 0	-0.5 ± 0.71	1 ± 0	1 ± 0	0.18 ± 0.94
	Asbru	0.76 ± 0.51	-0.13 ± 0.54	0.5 ± 0.71	-0.5 ± 0.71	1 ± 0	1 ± 0	0.36 ± 0.7
	Mean	0.7 ± 0.61	-0.29 ± 0.71	0.75 ± 0.5	-0.5 ± 0.58	1 ± 0	1 ± 0	0.27 ± 0.83
If Then Else	Clinical	1 ± 0	0.88 ± 0.5	0.5 ± 0.58	0.5 ± 1	1 ± 0	1 ± 0	0.88 ± 0.43
	Asbru	0.94 ± 0.25	0.81 ± 0.54	-0.25 ± 0.96	0.5 ± 1	0.17 ± 0.41	1 ± 0	0.67 ± 0.62
	Mean	0.97 ± 0.18	0.84 ± 0.51	0.13 ± 0.83	0.5 ± 0.93	0.58 ± 0.51	1 ± 0	0.78 ± 0.54
Plan Activation	Clinical	1 ± 0	0.78 ± 0.67			1 ± 0	1 ± 0	0.92 ± 0.41
	Asbru	1 ± 0	0.78 ± 0.67			1 ± 0	1 ± 0	0.92 ± 0.41
	Mean	1 ± 0	0.78 ± 0.65			1 ± 0	1 ± 0	0.92 ± 0.4
Parallel Plan	Clinical	0.64 ± 0.81	0.27 ± 1.01	0.5 ± 0.93	-0.25 ± 0.71	1 ± 0	1 ± 0	0.38 ± 0.88
	Asbru	0.82 ± 0.39	0.64 ± 0.5	0.13 ± 0.35	0.13 ± 0.35	0.5 ± 0.71	1 ± 0	0.51 ± 0.51
	Mean	0.73 ± 0.62	0.45 ± 0.8	0.31 ± 0.7	-0.06 ± 0.57	0.75 ± 0.5	1 ± 0	0.45 ± 0.72
Sequential Plan	Clinical	0.71 ± 0.73	0.71 ± 0.61	0.67 ± 0.58	-1 ± 0	0 ± 1	0.33 ± 1.15	0.51 ± 0.82
	Asbru	0.79 ± 0.58	0.71 ± 0.61	-0.33 ± 0.58	-0.33 ± 0.58	0 ± 1	0.33 ± 1.15	0.5 ± 0.75
	Mean	0.75 ± 0.65	0.71 ± 0.61	0.17 ± 0.75	-0.67 ± 0.52	0 ± 0.89	0.33 ± 1.03	0.51 ± 0.78
Simple Plan	Clinical	0.86 ± 0.52	1 ± 0	1 ± 0	0.84 ± 0.55	0.69 ± 0.75	0.85 ± 0.75	0.89 ± 0.47
	Asbru	0.89 ± 0.42	1 ± 0	0.13 ± 0.34	0.84 ± 0.55	0.54 ± 0.78	0.85 ± 0.58	0.68 ± 0.56
	Mean	0.88 ± 0.47	1 ± 0	0.56 ± 0.5	0.84 ± 0.54	0.62 ± 0.75	0.85 ± 0.66	0.78 ± 0.52
To be Defined	Clinical	1 ± 0	1 ± 0	1 ± 0	1 ± 0			1 ± 0
	Asbru	1 ± 0	1 ± 0	1 ± 0	1 ± 0			1 ± 0
	Mean	1 ± 0	1 ± 0	1 ± 0	1 ± 0			1 ± 0
Switch case	Clinical					1 ± 0	1 ± 0	1 ± 0
	Asbru					1 ± 0	1 ± 0	1 ± 0
	Mean					1 ± 0	1 ± 0	1 ± 0
Total	Clinical	0.81 ± 0.57	0.52 ± 0.83	0.88 ± 0.42	0.54 ± 0.79	0.74 ± 0.68	0.81 ± 0.6	0.7 ± 0.69
	Asbru	0.87 ± 0.41	0.62 ± 0.54	0.15 ± 0.48	0.63 ± 0.67	0.52 ± 0.68	0.83 ± 0.53	0.63 ± 0.6
	Mean	0.83 ± 0.5	0.57 ± 0.74	0.52 ± 0.58	0.6 ± 0.73	0.65 ± 0.68	0.82 ± 0.56	0.66 ± 0.65

Explanation regarding the declarative KRs

Notice that (Table 50) for all EPs in most of the KRs, the MQS of the Asbru measure was the same as the Clinical measure.

Notice also that the MQS of EP8 in all KRs of the *Context* KR class is rather low in both Asbru and Clinical measures and that the MQS of the Asbru measure is lower than the Clinical measure in the *Filter Condition* KR and in the *Abort Condition* KR for both EPs in the case of PID GL

Explanation regarding the procedural KRs

Surprisingly (Table 51), in the markups of EP1 and EP2 in the PID GL, and in both markups of EP5, the *Asbru* measure was higher than the *Clinical* measure across all KR types, including the total MQS. The MQS for the *Cyclical*, *Sub-Plan Parallel Plan* and *Sub-Plans Sequential Plan*, were especially low in both measures (mean MQS of 0.27 ± 0.83, 0.45 ± 0.72, 0.24 ±0.78, respectively), and the Mean MQS of *Asbru* was equal or greater then the Clinical measure.

When we performed the proportion test for the number of scores of 1 in the Asbru and the Clinical measures in each KR, we found non significant difference (P>0.05) of the proportions for the *Filter Condition, Abort Condition and Simple Action* KRs. See appendix D.4 for the complete results. Thus, in almost of the KRs there is no significant difference between the Asbru and Clinical measures.

❖ Explanation for all the results:

After reviewing the previous results(see result 5.3.1), it should not be surprising that the KRs of the *Intentions* KR class, which was first after sorting, are belongs the *Easy* group, and the KRs of the *Condition* KR class, which were ranked last after sorting, are belong to the *Medium* and *Difficult* group. The difficulty in the *Condition* KR class can be explain by the not detailed OSC the EP used in the PID GL (that is, deliberately lack of operator between the sentences of the condition, see 4.2.3 for explanation), and because the rather low completeness (36%) of EP8 in the COPD GL. The difficulty in the procedural KRs such as *Cyclical* , *Parallel Plan* and *Sequential Plan* can be explained by the difficulty of EPs to structure procedural complex structure

The low MQS of EP8 of both Asbru and Clinical measures in the *Context* KR class for the COPD GL (Table 50) can be explained by his low completeness measure. The lower MQS of the Asbru measure than the Clinical measure, and the significant difference between these measures in the *Filter Condition* and in the *Abort Condition* of both EPs in the PID GL can be explained by the less detail OSC they used.

The higher MQS of the Asbru measure than the Clinical measure in all procedural KRs of both EPs in the PID GL, and of EP5 in his both markups can be explained by the well structured OSC they used decreasing the Asbru semantic ambiguity (although the OSC of the PID was less detailed then the others OSCs), although they had still some ambiguity regarding the clinical semantics of the GL. Another explanation might be that the EPs didn't follow the OSC clinical directives, and paid more intention of the Asbru semantics. The significant difference in the Asbru and Clinical measures in the *Simple Action* KR can be explain by the low MQS of EP8 in the COPD GL because he structured most of the simple action as text sentences and not created a plan for them

❖ Conclusion :

EPs can perform markup with high proportion of scores of 1 for all KR classes and KR types and with high correctness of both Clinical and Asbru measure. Declarative KRs are easier to structure than the procedural KRs, thus a graphical interface for structuring the complex procedural KRs might improve their correctness measure. Perhaps an interface which supporting the EP in the clinical aspects, such as monitoring the EPs input might be more appropriate for this kind of KRs. Making a detailed OSC is crucial for achieving high correctness. It is more important when having an OSC that the editor will follow it in order to increase the correctness of both Asbru and Clinical measures.

5.3.3. Results regarding the correctness of the tasks

❖ Research question and its results:

(a) What is the Mean Quality Score of the *Application* and *QA* tasks in each markup of each EP, and for each GL? Is the proportion of scores of 1 (out of-1, 0, 1) significantly higher than 1/3 for each EP in each task? (b)Is there a significant difference in that correctness between the tasks for each markup of an EP and for all markups?

(For the complete hypothesis, see hypothesis 5 in section 4.6.3)

(a) What is the Mean Quality Score of the *Application* and *QA* tasks in each markup of each EP, and for each GL? Is the proportion of scores of 1 (out of-1, 0, 1) significantly higher than 1/3 for each EP in each task?

- Method of measurement:
  To measure the MQS for the tasks, formulas 38-42 (see section 4.4.5) were used. To test whether the proportion of scores of 1 significantly higher than 1/3 for each EP in each task, we performed a binomial proportions test, in which the scores of -1 and 0 were aggregated as one score versus the score of +1 for the whole task.
- The results are shown in the following tables:

Table 52. The Mean Quality Score of the tasks in each markup

		PID		COPD		HypoThyd		All
		EP1	EP2	EP8	EP5	EP8	EP5	Mean
Application	Clinical	0.85 ± 0.51	0.56 ± 0.77	0.7 ± 0.69	0.56 ± 0.79	0.67 ± 0.76	0.73 ± 0.69	0.68 ± 0.69
	Asbru	0.71 ± 0.5	0.47 ± 0.62	0.1 ± 0.59	0.63 ± 0.68	0.47 ± 0.74	0.64 ± 0.68	0.53 ± 0.63
	Mean	0.78 ± 0.51	0.51 ± 0.7	0.4 ± 0.71	0.59 ± 0.74	0.57 ± 0.75	0.68 ± 0.68	0.61 ± 0.67
QA	Clinical	0.86 ± 0.5	0.56 ± 0.81	0.42 ± 0.89	0.58 ± 0.77	0.82 ± 0.58	0.84 ± 0.54	0.67 ± 0.73
	Asbru	0.9 ± 0.36	0.62 ± 0.66	-0.08 ± 0.69	0.64 ± 0.68	0.66 ± 0.61	0.77 ± 0.54	0.61 ± 0.67
	Mean	0.88 ± 0.44	0.59 ± 0.74	0.17 ± 0.84	0.61 ± 0.73	0.74 ± 0.6	0.81 ± 0.54	0.64 ± 0.7
ALL	Clinical	0.88 ± 0.46	0.58 ± 0.76	0.34 ± 0.93	0.59 ± 0.77	0.76 ± 0.66	0.8 ± 0.61	0.66 ± 0.72
	Asbru	0.77 ± 0.46	0.49 ± 0.65	-0.09 ± 0.73	0.64 ± 0.7	0.61 ± 0.67	0.73 ± 0.6	0.54 ± 0.68
	Mean	0.82 ± 0.47	0.54 ± 0.71	0.13 ± 0.86	0.61 ± 0.73	0.68 ± 0.67	0.77 ± 0.6	0.6 ± 0.7

Notice that (Table 52) the mean MQS scores were quit high for all EPs in the two tasks (except EP8 in the COPD GL in the Application task): The mean MQS of the *Application* task was 0.61 ± 0.67, and the mean MQS of the *QA* task. was 0.64 ± 0.7.

In the *Application* task there seems to be variability between the EPs in the MQSs with range of [0.1,0.78]. In the *QA* task, except for EP8 in the COPD GL, all EPs achieved higher MQS than in the *Application* task.

In addition, it can be seen that the MQS of EP8 in the COPD GL is quite different between the tasks: in the *QA* he achieved MQS of 0.17 ± 0.84 and in the *Application* task his MQS raises to 0.4 ± 0.71.

Table 53. The proportion of scores of 1 for each markup of an EP in each task, and for all markups

	PID		COPD		HypoThyrd		
	EP1	EP2	EP8	EP5	EP8	EP5	All
Application	82.18%******	63.16%**	52.86%*	74.29%****	72.22%***	89.04%******	71.17%****
QA	85.6%******	66.24%***	43.75%*	76.29%*****	79.59%*****	91.92%******	72.72%*****

* denotes significant (P<0.05) for proportion of scores of 1 > 0.3: ** for proportion >0.5; *** for proportion >0.6; ****for proportion >0.65; *****for proportion >0.7; ******for proportion >0.75.

Notice that (Table 53) the proportion of scores of 1 in all markups of all EPs in both tasks are significantly (P<0.05) higher than 0.3, and in most of them higher than 0.5, 0.6, and even than 0.75.

(b) Is there a significant difference in that correctness between the tasks for each markup of an EP and for all markups?

- Method of measurement:

To test weather there is significant difference between the tasks for each EP we performed Proportion test for the number of scores of 1 between each pair of tasks of an EP and for all EPs

- . The results are shown in the following tables:

Table 54. The result of the proportions of scores of 1 between the task for each EP, and for all EPs

		Application		QA		z statstic	P value	Confidence Interval (95%)
		Proportion	N	Proportion	N			
PID	EP1	82.18%	249	85.60%	321	1.21 *	0.226	[-0.09 , 0.022]
	EP2	63.16%	204	66.24%	259	0.86 *	0.391	[-0.101 , 0.04]
COPD	EP8	52.86%	74	43.75%	84	1.64 *	0.101	[-0.2 , 0.017]
	EP5	74.29%	104	76.29%	148	0.42 *	0.675	[-0.114 , 0.074]
HypoThyrd	EP8	72.22%	52	79.59%	78	1.12 *	0.263	[-0.204 , 0.057]
	EP5	89.04%	65	91.92%	91	0.64 *	0.521	[-0.118 , 0.061]
ALL		71.97%	896	72.72%	981	0.43 *	0.668	[-0.042 , 0.027]

* denotes a non significant (P>0.05) between two proportion of score of 1 between the two tasks

Notice that (Table 54) there is no significant difference between the tasks in the proportion of scores of 1 for each markup of an EP and for all markups. The results also showed that for all the markups the confidence interval around the mean difference between the proportions was very narrow. In other words, the statistical power of the test was adequate

❖ Explanation for actual results:

In table 52 the lower MQS of the *Application* task than the *QA* task can be explained by the fact that as we saw in Hypothesis 3 (see section 5.3.2), the MQS of the *Conditions* KR class was slightly low for all EPs (MQS of 0.45± 0.69).The *Application* task which composed mainly from this KR class and not from KR classes with higher MQS (such as *Context* and *Intentions*). The higher MQS of EP8 in the COPD GL in the *Application* task than in the *QA* task can be explained by its low MQS in the *Context* KR class (-0.83 ± 0.57) and its low completeness level (only 8%). The low MQS of the COPD in the *QA* task and in the general mean can be explained, by the low MQS of EP8 (0.13±0.86). Because the common problem in the *Conditions* KR class among all EPs, the MQS of *QA* task was higher than the *Application* task, in all GLs (except the COPD, again because the low MQS of EP8)

The non significant difference between the proportions of scores of 1 between the tasks can be explained by the fact that the *Plan-Body* KR class is common for both tasks.

❖ Conclusion:
Using an explicit OSC, EPs can perform markup with the same proportions of correctness for both tasks: the *Application* task and the *QA* task. Thus, as mentioned in the previous conclusions, graphical interfaces supporting acquisition for the more difficult KR classes such as *Conditions*, will increase its MQS for each task as well.

5.3.4. Results for type of errors

❖ Research question and its results:
(a) What is the proportion of the general errors (see section 4.4.2) between the Clinical and Asbru measures in each GL and for all GLs? (b) What is the number of each type of error in the general errors scale for both Asbru and Clinical measures? (c) What is the number of specific errors (see section 4.4.2) of each KR for each EP, and for all EPs?
(For the complete hypothesis, see hypothesis 4 in section 4.6.3)

(a) What is the proportion of the general errors (see section 4.4.2) between the Clinical and Asbru measures in each GL and for all GLs?

- Method of measurement:
  For calculating the proportion, we counted and summarized the number of general errors in each measure for each GL and for all GLs
- The results:

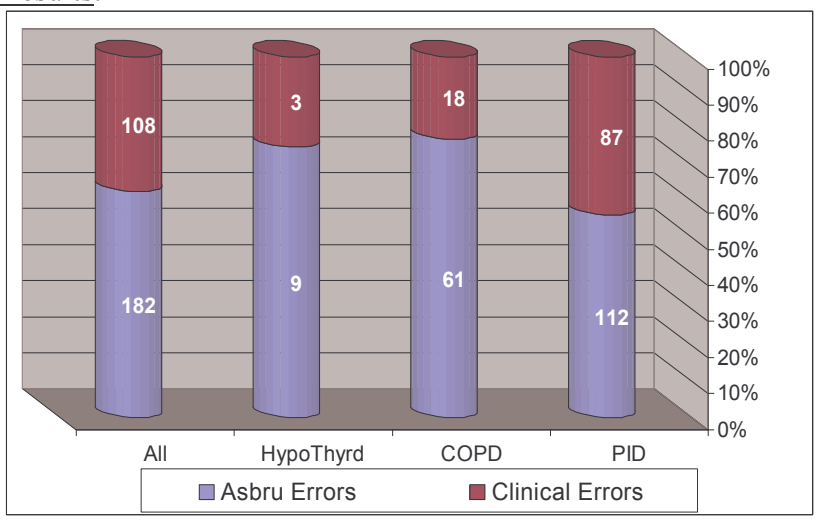


Figure 35. proportion between the Clinical and Asbru general errors (see section 4.4.2) in each GL and for all GLs

Notice that (Figure 35) the highest number of both Asbru and Clinical errors are presented in PID GL with same proportion. Then, the COPD has less errors and the proportion of the errors in the Asbru measure is rather higher then its Asbru measure. Finally, the HypoTyrd has the lowest amount of errors and with the same proportions as in the COPD. Altogether, there were 182 reported Asbru errors and 108 Clinical errors.
However, there were some exceptions for assigning errors: sometimes even when EP had some error, he was assigned a "correct" score, this is because his error was not so worse, but still we wanted to have some documentation of this type of error. For example, the error of incomplete labeling the of drug name in the plan name, but the drug name appears in the textual content of the plan.

(b) What is the number of each type of error in the general errors scale for both Asbru and Clinical measures?

- Method of measurement:
  For calculating the number of errors, we counted and summarized the number of errors for each type of error in both, the Asbru and the Clinical measures.
- The results:
  The results for both Asbru and Clinical measures for each type of error in each GL and for all GLs are detailed in the following figures in the next page (Figures 36, 37). For more detailed results of errors see Appendix D.5.

Types of errors in the Asbru measures

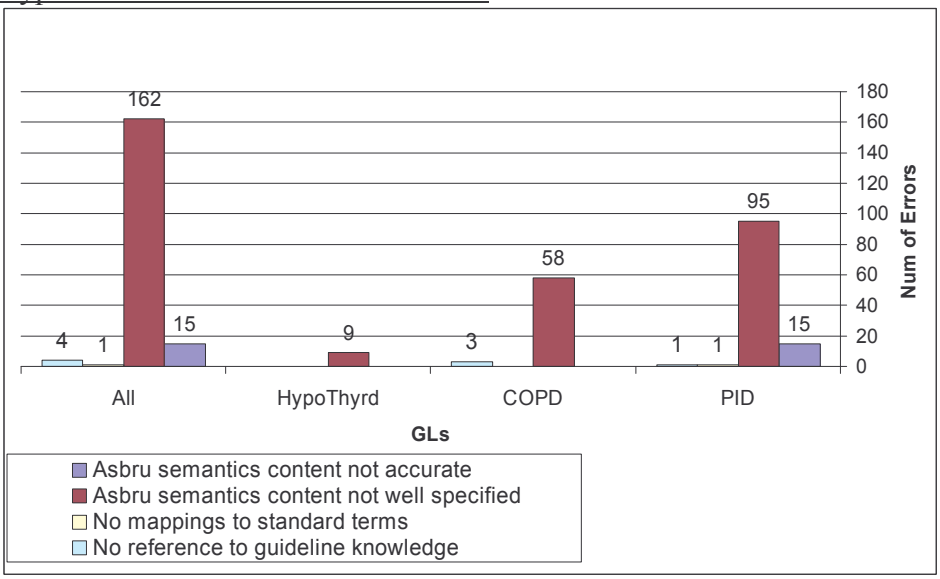


Figure 36 The types of errors in the Asbru measures in each GL and for all GLS

We can notice that the error type of "Asbru semantic content not well specified" was the major error type for all GLs. However, this error is decreasing among the GLs

Types of errors in the Clinical measure

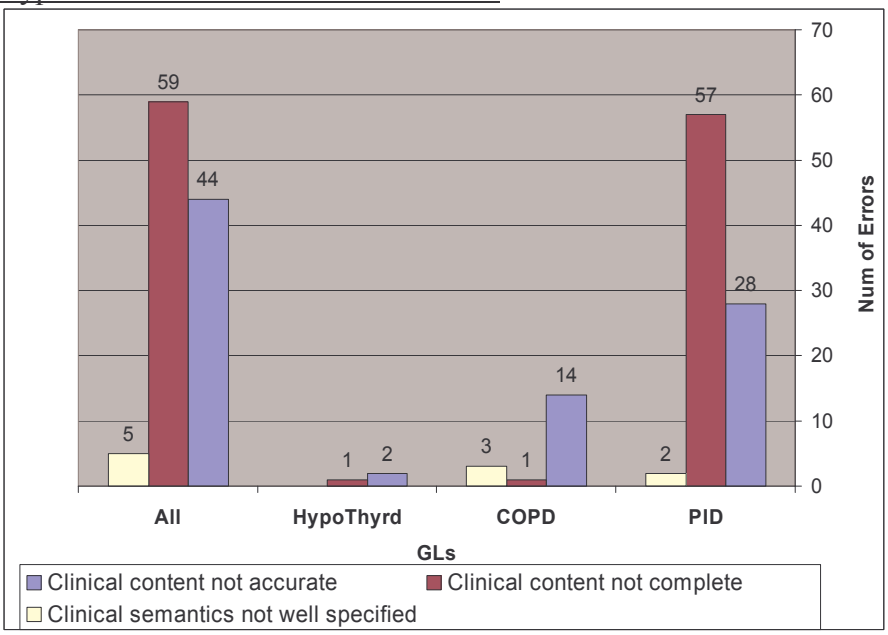


Figure 37. The types of errors in the Clinical measures in each GL and for all GLS

Notice that (Figure 37) the error type of "clinical content not complete" common to all GLs , but was the major error in the PID which might be explains the higher MQS in the Asbru measure in this GL. However, this error is decreasing almost to zero in other GLs. In addition, the error of "clinical content not accurate" is common for all GLs, but decreasing among them.

(c) What is the number of specific errors (see section 4.4.2) of each KR for each EP, and for all EPs?

- Method of measurement:
  For calculating the number of errors, we counted and summarized the number of specific errors in KR for each EP , and for all EPs.
- The results:

Table 55.the number of specific errors for each KR for each EP and for all EPs

			PID			COPD			HypoThyrd		
	KR Type	Error Description	E1	E2	ALL	EP8	EP5	ALL	EP8	EP5	ALL
Conditions	Filter Condition	There are no And/Or operators between the different criteria's	21	31	52						
	Abort Condition	There are no And/Or operators between the different criteria's	7	5	12						
Plan-Body	cyclic	"starting specification" is incorrect	1	1	2	1		1	1		1
		"starting specification" is missing		14	14						
		"repeating specification" is not well specified		13	13				1		1
		"repeating specification" content is missing				1		1			
		The plan has incorrect type				1	1	2			
	IfThenElse	There is no pointer for GK in the condition		1	1	6		6			
		The plan has incorrect type				2		2			
	parallelPlan	Not all the necessary sub -plans are further defined)	1		1						
		Some of the sub-plans are not mandatory when they should	1	3	4	3	7	10			
		Appears as "simple action" in the markup				4		4	1		1
	sequentialPlan	Some of the sub-plans are not mandatory when they should	1	1	2	1	2	3	1		1
		Some of the sub-plans has incorrect type	1	1	2	1	1	2			
		Some Subplans are missing				1		1			
	simpleAction	The Simple Action text is not describing well the action itself	19	25	44						
		Appear as text content of in the markup				32		32	2		2
			52	95	147	53	11	64	6		6

Notice that (Table 55) all errors of the *Conditions* KR class in the case of PID GL are decreasing to zero in the other GLs. The same for most of the errors of the *Plan-Body* KR class: they decrease almost to zero between the PID and the COPD GLs and between the HypoThyrd GL

❖ Explanation for all the results:

There is a clear trend: The amount of errors are highest in the PID GL because its relatively highest number of plans (106 plans) and its less detail OSC. The number of errors decreases as the OSC becomes more detailed (COPD GL), and reaching almost to zero in the HypoThyrd GL, which uses the most detailed OSC. Notice that the deliberately errors in the *Conditions* KR class in the PID OSC decreases to zero in the other GLs because the operators of the condition were included in their OSCs. The same for the error of the *Sub-Plan Parallel* and *Sequential* KRs with the description of "some sub-pans are mandatory when they shouldn't": The mandatory plans were not denoted explicitly in the PID and the COPD OSCs and

therefore there were errors of this type. But, when it was denoted explicitly in the Hypothyrd OSC the errors decreases to zero.

Thus, our hypothesis confirmed: the EPs had more errors in the Asbru measure than in the Clinical measure, and the amount of errors decreased as the OSC was more detailed: The highest amount of errors was in the PID which used the least detailed OSC, fewer errors were in the case of the COPD GL which used more detailed OSC, and finally the least amount of errors was in the case of the HypoThyrd GL , which used the most detailed OSC.

- Conclusion :

  Making a detailed OSC is crucial for achieving high correctness and low amount of general and specific errors. Graphical interfaces should be used to decrease the amount of the errors in the Asbru semantic.

Summary

Analyzing the objective results, we found that EPs can perform markup with high, positive correctness for all KR classes and KR types, for both Clinical and Asbru measures, and for both *Application* and *QA* tasks. The declarative KRs were easier to structure than the procedural KRs, therefore a graphical interface for structuring the complex procedural KRs might improve the correctness measure among EPs and help them to bridge the gap between the initial structuring of the GL and the semantics of the specification language. Perhaps an interface which supports structuring of the clinical aspects, such as monitoring the EPs input might be more appropriate for decreasing the clinical errors as well.

Making a detailed OSC is preliminary crucial step for achieving high correctness. It is essential when having an OSC that the editor will follow it in order to increase the correctness of both Asbru and Clinical measures. Finally, it might also be conjectured that a less experienced EP with better computational skills could better follow the OSC. The comprehension of the target ontology might be more important.

6. A New Graphical Guideline Specification Framework – GESHER

6.1. The GESHER System and its philosophy

In parallel to the stage of getting the results regarding the URUZ low usability, its lack of functionality, its cumbersomeness and the fact that is not intuitive for eliciting procedural structures such as the tree of plans using the PBW (see section .4.2), we realized we should developed a more intuitive framework which should be graphical oriented and enable specification in multiple representations.
When designing a multiple representation-format (hybrid) specification tool, we have to keep in mind that there are several important requirements to attend to:

1. The tool should be able to access centralized resources (such as the DeGeL library) as a client application.
2. The tool should be independent of any medical domain.
3. The tool's architecture must support specification at multiple representation levels.
4. The tool should enable collaboration between two types of users: EPs and KEs.
5. The tool should enable acquisition of both procedural and declarative knowledge.
6. The tool should be able to handle multiple specification languages (GL ontologies).
7. The tool should provide authentication and authorization services (directly or through the overall framework within which it is used).
8. The tool should be able to access knowledge bases (KBs) in order to store and use the procedural and declarative knowledge of the GL.
9. The tool should use standard medical vocabularies for referring to medical terms, to enhance reusability of the GL's edited by the tool.
10. The tool's interface must be user friendly and highly usable.

6.1.1. The GESHER System

We have designed and implemented a new system for GL specification in multiple representation Levels (***GESHER***[11]). GESHER is a graphical client application, developed with Microsoft Dot.Net WinForm technology (Figure 36), thus answering desideratum 1. GESHER supports the gradual specification process of the GL according to DeGeL's hybrid GL representation model which is not dependent on any particular medical domain, thus answering desideratum 2. The GESHER system manages the specification process which incrementally structures a GL at multiple representation levels, according to the chosen target ontology. Each target ontology composed from KRs whose semantics are determined by the specification ontology (e.g., filter condition KR in the case of Asbru ontology).

[11] GESHER means " bridge" in Hebrew and stands for the bridge between the expert physician and the knowledge engineer in the process of guideline specification

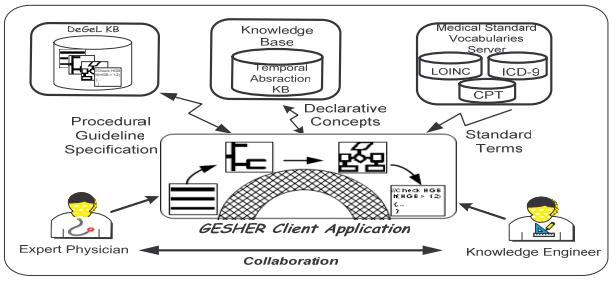


Figure 38. The GESHER Architecture. Expert physicians and knowledge engineers collaborate on incrementally specifying the guideline. GESHER uses DeGeL's knowledge base to save the procedural guideline specification at the different representation levels; The temporal abstraction knowledge base to save declarative concepts relevant to the guideline; and the standard vocabularies server to include standard terms

During this incremental process, several of the GL's KRs might exist at different levels of specification. In addition, the GL itself might be decomposed into sub-guidelines that are candidate for further specification. The specification task is best handled within one integrated framework. Thus, GESHER uses DeGeL server (Figure 36) for managing, storing and accessing the relevant procedural knowledge at all levels during the specification process, thus answering desideratums 3 and 8. GESHER uses graphical interface, enabling the gradual specification process be accomplished by an EP with relatively little training, by KE, or by both collaborating together. EPs are usually involved in the semi-structuring phase as they are familiar with the clinical knowledge inhered to the GL. KEs are usually involved with the semi-formal and formal phases as they are familiar with the GL specification language (the underlying target GL ontology). Sometimes, both EP and KE may work together (see figure 36), especially on the semi-formal level, thus answering desideratum 4. GESHER provides specification tools for the different types of users in all stages of the specification process, allowing capturing the GL's semantics and enabling EPs and KEs to define its procedural and declarative knowledge, thus answering desideratum 5.

The task of using multiple ontologies in GESHER is supported by DeGeL's hybrid meta ontology [Shahar and Young 2004].which supports specification of multi-ontologies of GLs, at least at the semi-structures level, using multiple target ontologies such as GEM or DeGeL's default ontology, Asbru. The meta ontology specifies the KRs common to all GL ontologies (e.g. documentation, classification indices) and the format in which any ontology is described, thus answering desideratum 6. The knowledge editing authentication model is taken very seriously and is handled by DeGeL's authorization and authentication model- DeGeLock [Shahar and Young 2004; Shahar, Shalom et al. 2003].

After a user is authenticated, her profile is retrieved. The user's profile contains only the tasks she is permitted to perform in GESHER (and implicitly on DeGeL's KB), thus answering desideratum 7. Some of the of the GL's specification is of a declarative type. For example, in the semi-structured level, an EP can find a description of what is "high HGB state" in a particular context.

The concept "HGB State" is an example of declarative knowledge, which should be defined in a formal format according to its various allowed states: High, Normal or Low. For this task, or when needed to define complex expressions (e.g. "Low HGB State for 2 weeks"), GESHER is linked the temporal abstracted knowledge acquisition tool (TAKAT) (see figure 36), which is one of the tools used within the IDAN architecture [Boaz and Shahar 2005]. For simple expressions (e.g. "HGB > 7.8

gr/dl"), the GESHER system uses the Expression Builder module (see section 3.2.2), thus answering desideratum 8.
When EPs or KEs need to define a clinical term (e.g., "serum hemoglobin"), they can select standard term from a medical vocabulary. Standard terms are defined using several controlled medical vocabularies and can be searched and retrieved using the MEIDA [German and Shahar 2005] system (see figure 36), which includes a vocabulary server and a search engine. The use of standardized vocabularies and terms enables execution of queries in the IDAN framework by the Spock GL runtime application system [Young 2005], regardless of the terminology used in each local clinical DB, thus answering desideratum 9.
Using the highly intuitive, user friendly, graphically oriented interface of the GESHER system, the specification process becomes smooth for both the EP and the KE. By developing specific graphical widgets for acquisition of each KR at each representation level, GIEML answers the needs of each one of the user types. For example, the methods needed by the EP specifying semi-structured knowledge (e.g., widget for markup of free text) are significantly different from the methods needed by the KE (e.g., widget for visual programming), thus answering desideratum 10.

6.2. The GESHER Interface

The GESHER architecture integrates several tools. The main one is a graphical tool for guideline mark-up, editing and specification in multiple representation levels, which facilitates the collaboration between the EP and the KE.
The main interface of GESHER system is presented in figure 37: with GESHER an EP can create a new GL document (GLDoc) according to one of the target ontologies available in DeGeL (e.g., GEM, Asbru). Then, one or more guideline sources (GLSrc), are selected from the DeGeL GL library using the Vaidurya GL search and retrieve engine [Moskovitch, Hessing et al. 2004], and portions of it's free text are labeled as a starting point (Figure 37). The smooth process of specification is enabled by the *Hybrid Ontology Tree* (HOT). HOT is composed from the KRs of the selected target ontology and has three different displayed views according to the three representation levels: semi-structured, semi-formal and formal view. The KRs which composing the HOT might be in different specification levels and are depending on the displayed view. For example, the KR *obtain values* is relevant to the semi-structured and semi-formal levels and therefore will be display when viewing those two views, while formal KRs such as Asbru's *returns* or *arguments*, are relevant to the formal level and therefore will be display in the formal view only. The different representation levels for every KR are implemented in the HOT as nodes and sub nodes. Each level is a sub-node of the parent KR node. Thus, for the same KR, there are three different sub nodes, a node for each representation level. For each KR at a representation level, a graphical widget is generated on the fly (Figure 37).

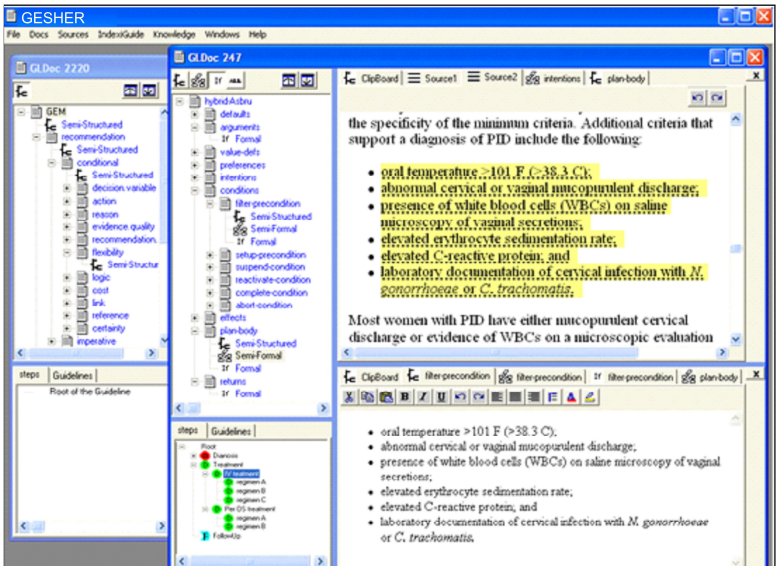


Figure 39. The GESHER system's main interface. Two guidelines documents are shown, each with a different target ontology: GEM (background window) and hybrid Asbru (front window). In the hybrid Asbru front document, the hybrid ontology tree appears in the upper left, showing the semi-structured, semi-formal and formal views. The guideline source is open in the upper view area and the semi structured filter-condition knowledge role (KR) (along with other KRs) is open in the lower design area, containing portion of the labeled text, dragged from the source

This widget is added to the upper *view* area or to the lower *design* area, enabling specification of the selected KR according to the selected representation level.
Currently, we support the semi-structured level for multiple GL ontologies (e.g GEM, Asbru), and the semi-formal level for the default ontology used in DeGeL, Asbru. Thus, we have created the *Hybrid Asbru* ontology which embeds the semi-structured, semi-formal and formal Asbru semantics in one hybrid ontology.
The semi-structured view is implemented as a graphical widget generated for any KR in the semi-structured representation level (in any target ontology). This widget contains HTMLEditor frame which enables the user to perform semi-structured mark-up by dragging a portion of labeled content from one or more GLSrcs (this content may be text, tables of figures) into a selected KR's frame, and perform manipulation on this text with reach design toolbar. The positions of this markup at each GLSrc are saved and the labeled text is highlighted (see figure 37).
This process enables turning implicit knowledge into more explicit fashion by the EP, facilitating the task of the KE (who is not familiar with the GL semantics) towards fully formalizes the guideline.

6.3. The Semi-Formal Representation View

Structuring GL at the semi-formal level requires developing designated widgets with intuitive interfaces to handle the acquisition accordance KR semantics. For example, we implemented a semi-formal view for semi-formal Asbru. Semi-formal Asbru is a simplified version of Asbru, with similar semantics to the full version, but with somewhat less complex syntax. The main reason we use semi-formal Asbru is to improve the collaboration between the EP and the KE during the GL specification process, especially after an EP semi-structured the GL and before a KE semi-formalized it. Semi-formal Asbru has most of Asbru's KRs such as plan-body which embedded the procedural knowledge of the GL and acquired with the Hierarchical Plan Builder Tool (see section 6.3.1), and conditions (e.g. eligibility completion, and filter condition) for describing time-annotations and simple temporal constraint which are acquired with the Expression Builder Tool (see section 6.3.2).

6.3.1. The Hierarchical Plan Builder

The semi-formal representation level is usually specific to each selected GL ontology, therefore, we have implemented the Hierarchical Plan Builder (HPB) (Figure 38), which is customized for the procedural aspects of the hybrid Asbru ontology (e.g., the plan-body KR in the case of the hybrid Asbru ontology). However, most of GL ontologies describe the procedural knowledge as a hierarchy of plans and sub plans, thus, this tool is actually quite generic

The HPB facilitates the task of decomposing the GL into sub-guidelines in a transparent process: The first step towards decomposing the GL into sub-plans is very straightforward and usually performed by the EP by specifying in general fashion the GL structure using an initial plans with basic semantic type (Figure 38). This type can be Intervention (e.g., *Education*, *Procedure*, or *Drug* types) or more general as *Observation* or *Follow-Up* types. In addition, we allow *General* plan, which its type will define later. For referring a pre-defined GL in DeGeL, *Public-Ref* plan can be created. When needed control elements, plans with semantics of *periodic*, *condition,* or *Switch-case* can be defined. Each plan is candidate for decomposing into sub-plans. When it decomposed, a new sub-level is created with at least two plans which are might further decomposed as well. In addition, control structure of ordering (e.g. sequential, parallel) might be added for each group of plans in the same level (Figure 38). A labeled portion of text with the plan's description can be related to each plan from its parent plan textual content

In the next step, which usually performed by the KE, or together with the EP, the user further specifies the plans and its sub-plans into semi-formal format (semi-formal Asbru in this case). For example, each plan with initial semantic type (e.g., drug, education) is *"To-Be-Defined"* plan in semi-formal Asbru. A decomposed plan specified as "*Subplan*", periodic plan as "*Cyclical*", and conditional plan as "*if-then-else*" in the semi-formal Asbru. Condition and switch-case plans have an expression as part of their semi-formal Asbru semantics. This expression can be specified in a semi-formal format using the Expression Builder tool (see section 3.2.2).

Thus, the HPB is a bridge between the EP and the KE, enabling collaboration towards the semi-formal and formal representation levels.

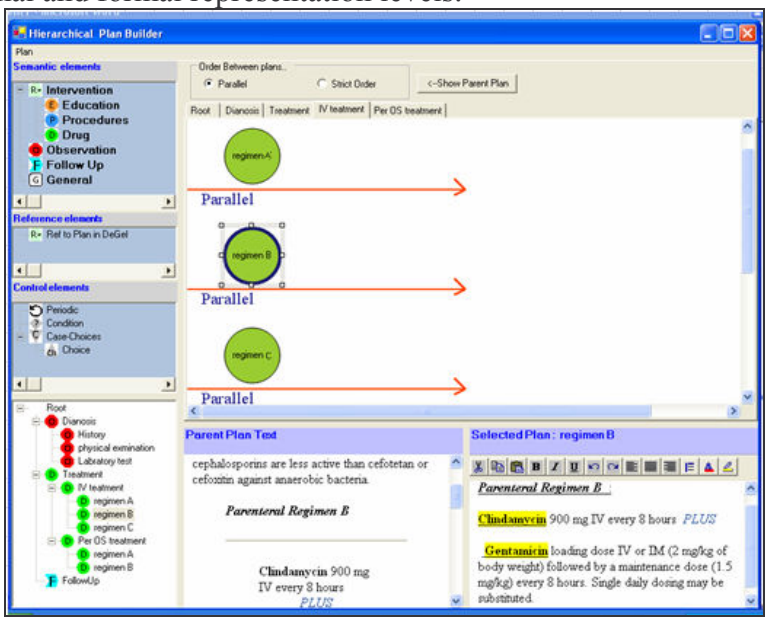


Figure 40. The Hierarchical Plan Builder in GESHER, showing how the procedural aspects for the guideline being specified. In this case, a plan for IV treatment of the Pelvic Inflammatory Disease guideline, which specifies three different regimens that should be performed in parallel

6.3.2. The Expression Builder

Conditions (e.g., filter, complete) are one of the most important KRs in hybrid Asbru. A condition is a boolean expression which is evaluated to *true* or *false*. An example of using condition as filter condition is when trying to formalize the sentence "a pregnant woman whose hemoglobin level is greater then 20.gl/dl". The graphical widget used within the GESHER system for defining conditions is called the "Expression Builder" and is shown in figure 39. IDAN query language [Boaz and Shahar 2005] used as the default formal language for temporal queries, but we could also use other standards such as HL7 RIM [HL7-RIM] by manipulating the output of the Expression Builder.

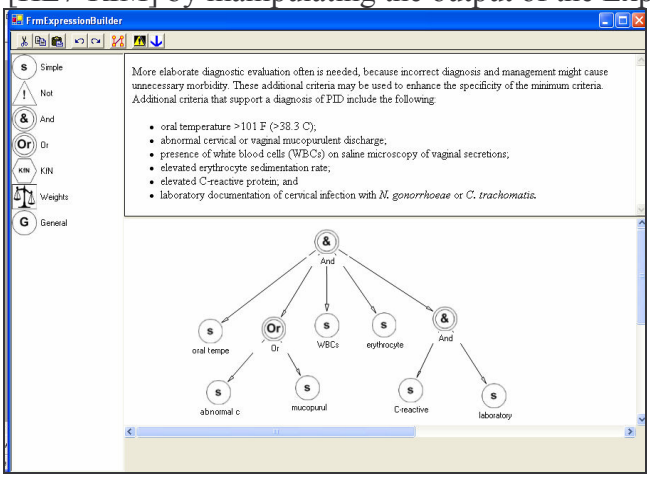


Figure 41. The Expression Builder in GESHER Using pre-defined condition types (e.g, simple, boolean expression) the user specifies semi-formal expressions. The semi-structured content of filter condition knowledge role (see figure 37) is being specified in the semi-formal level

6.4. The Formal Level Representation View

Our research focused mostly on the semi-formal level, which is the main phase in the GL specification process. After this phase most of the GL specification was defined and the KE can continue to the final step thought formal representation level.
Nevertheless, work was done towards full-structured level: expressions, time annotations and simple temporal patterns which are conditions can be expressed with the Expression Builder tool and thus enabling execution in the IDAN by Spock system. Most of the procedural (e.g., plan-body KR) and declarative knowledge (e.g., the various conditions KRs) can be define according to hybrid Asbru semantics.

Summary

We have presented GESHER, a client application graphical tool for specification of clinical GLs in multiple representations and in multiple GL ontologies. By using the GESHER tool, which is independent of any particular medical domain, the specification process of the GL becomes smooth and transparent, enabling collaboration between EP and KE. Using previously defined terms, GESHER can be characterized as both a *document-centric* and a *model-centric* tool, since it supports both a bottom-up ontology-driven semantic markup of the source document (mostly for declarative knowledge roles), as well as a top-down construction of the procedural body of the GL. GESHER uses DeGeL as the procedural knowledge server allowing knowledge of any specification level in any different representation level to be store and re-use. Temporal abstracted knowledge base enables declarative specification of the GL. Finally, terms from standard medical vocabularies can be searched and use in the GL by using the MEIDA search engine. The GESHER tool allowing specification

using different graphical widgets for different KRs in any displayed view, and includes generic semi-structured widgets and Asbru-specific semi-formal and formal widgets.

Ongoing evaluation is currently conducted with our collaborators with preliminary encouraging results. Future work will focus on the formal representation level. We aim to develop additional graphical widgets to support the specification of formal KRs. In addition, we expect GL ontology designers to develop additional widgets to support the semi- the semi-formal and formal views of other specification languages. In addition creating an OSCs using GESHER is one of the things to be considered when developing it.

7. Summary, Conclusions and Discussion

7.1. Summary

Care providers, overloaded with information, rarely have the time, or the computational means, to use the valuable knowledge encoded in clinical guidelines (GLs), during treatment. In addition, although there are many textual GLs, most are not automated. There is therefore a pressing need to facilitate automated GL specification, dissemination, application, and quality assessment. Furthermore, there is a need for a well-defined methodology that integrates the roles of expert physician (EPs) and knowledge engineer (KEs), and that supports specification and conversion of the GL's free-text representation by EPs, through semi-formal representation into a machine comprehensible representation, thus enabling automated support. Thus, a hybrid representation model was therefore developed by us, which combines optimally the relative skills of the KE and the EP. This hybrid model was implemented in an architecture and set of tools, called the Digital electronic Guideline Library (DeGeL) which was developed to support GL classification, semantic markup, context-sensitive search, browsing, run-time application, and retrospective quality assessment. One of the DeGeL framework tools is the web-based URUZ markup tool. As a web-based tool as part of DeGeL framework, using the infrastructure of the hybrid guideline representation model, URUZ enables EPs and KEs in different sites to collaborate in the process of GL specification and mark up the GL in any representation level using *Asbru* language as the underlying guideline-representation language.

In this study, a methodology for the overall specification process of GLs from a textual representation into a semi-formal representation was developed and evaluated. Three different GLs from three different clinical domains were used: *Pelvic inflammatory disease* (PID) in the Gynecology domain, *Chronic Obstructive Pulmonary Disease* (COPD) in the Pulmonary domain, and *Hypothyroidism* in the Endocrinology domain. It should be noted that each GL, in fact, includes multiple subguidelines specialized for various contexts. Eight EPs and two KEs participated in this overall process. Each GL was marked-up by two EPs. For each GL, a Gold Standard (GS) was made by a different EP, with the collaboration of a KE. The specification and evaluation of all markups took around six monthss

7.2. Conclusions and specific contributions

As a result of this research, four main insights for improving the GL specification process can be presented. These insights fall into four main categories:

1. The creation of an Ontology Specific Consensus;
2. The essential aspects that need to be learned to support the specification process;
3. The medical and computational qualifications needed for specification of a GL;
4. The characteristics of the KA tool needed to support semi-formal specification of a GL.

The following subsections describe in detail each of the insights

7.2.1. Creation of an Ontology-Specific Consensus (OSC)

This research study has clearly shown that creating an OSC is an essential step before the GL specification process. Before using the detailed OSC, the researcher first tried using an informal consensus, without making any progress in the markup phase due to the multiple ambiguities and missing information inherent in most clinical GLs.

Creating an OSC forced the EPs to disambiguate many of the core consensus issues. It was also obvious that each markup editor would create a different version of the same GL, and the comparison to a GS would be meaningful. Thus, in this research, three different OSCs (one OSC for each GL) at an increasing level of detail were created. Although all of the markups were rather complete (mean weighted completeness of 91%), it was found that the more detailed and structured the OSC was, the less variability existed between the EPs in the correctness measure(see results in 5.3.2), and the fewer were the semantic errors (see results in 5.3.4). Thus, it is suggested that the OSC be made as detailed as possible, including all relevant procedural and declarative concepts. This approach has a potential drawback: at the extreme of this approach, the OSC will become very close to the markup itself. However, we believe that in most cases, the OSC will only serve the editors as a very rough draft representing the clinical semantics of the "spirit" of the GL specification in an ontology-specific fashion.

Regarding the aspect of who should create the OSC: as will be discussed in the third insight, it is suggested that the OSC should be created (as described in the research methodology in section 4.1.4) initially by a group of EPs of the local medical setting in collaboration with a KE(s) who is familiar with the specification language. The KE should teach the EPs the core semantics of the specification language in order to create "brain storming" not only between the KEs, but also between the EPs and the KEs. In our case, the EPs were integral participants in the decisions regarding the semantic specification of the GL which contributed to the expressiveness of the OSC, led to fruitful discussions, and created additional motivation among the EPs, especially among the EPs who were intended to do the markup. After a clinical consensus among the EPs exists, a single senior EP may continue working together with the KE in the remaining stages of converting the consensus into an ontology-specific one.

Another important aspect is that the OSC is independent of the specification tool; thus the OSC can be used for markup with different specification tools for the same ontology. Thus, an OSC such as was created here can be used when working with another specification tool that can support perhaps (among other ontologies) the Asbru ontology, such as the GESHER GL specification tool.

Finally, since making an OSC is a preliminary necessary step towards markup, and includes multiple cognitive aspects, such as interactions between the EPs and the KEs, this phase might involve psychologists, to explore the issue of "group thinking," in order to discover how best to benefit from this session. Another suggestion which is described in detail in the fourth insight is to use graphical tools to facilitate the OSC creation and specification in multiple languages; thus, the researcher proposes creating additional OSCs in other specification ontologies (for example the PID GL in the GLIF ontology). In addition, it is proposed that re-using and sharing of OSCs among EPs and clinical settings should be supported by saving the OSCs in an appropriate digital library. DeGeL might support this task.

7.2.2. The essential aspects needed to learn to support the specification process by EPs

It was found in this study that creating an OSC and performing the markups are two different tasks which require teaching two different aspects: creating an OSC requires teaching the EPs the overall representation framework (DeGeL in this case), and performing markup requires teaching them the GL specification tools (such as URUZ and IndexiGuide). However, learning the main knowledge roles of the specification

language is essential for both the tasks, and therefore should be considered the first step towards the specification. (This was clearly demonstrated in the analysis of the subjective questionnaires - see results in section 5.1)
It was also found that there was a group of knowledge roles (KRs) that were more difficult to understand, as reported in the questionnaires by the EPs, such as the *Intentions* and the *Guideline knowledge* KRs (see results 5.1.3). There was also a group of KRs that all the EPS found difficult to structure, such as the condition KRs *Complete, Abort and Filter* and the procedural KRs *Switch case*, *Cyclical*,and *Sub-Plan* KRs (see results 5.3.2). Bearing this in mind, in addition to the finding that when the EPs considered a KR of the specification language easy to understand, they also found it easy to structure (see results 5.1.6), it is suggested that in order to make the structuring process more intuitive, more effort should be invested in teaching these KRs in particular. Since the difficulty of acquiring procedural knowledge was realized during the process, the GESHER tool, which is a new graphical interface for GL specification and is described in the fourth insight, might be more helpful.
In addition, it was found that the EPs specify with no significant different groups of KRs for a *Q*A task than for an *Application* task; thus we can use their markup for quality assurance or application of a GL.
In addition, to make sure they understand the main KRs well, it is suggested that a short test should be administered to measure the EP's level of understanding of the specification language and of the markup task, before they perform markups. A help manual or a small simulation of marking up a GL could be used to assist the EP in the specification process to refresh the meaning of several concepts or to confirm understanding about a task related to the specification language, the specification tool, or generally about the specification process.
Finally, it is suggested that the participants in the evaluation session (see next insight) should learn to use the evaluation tool, in order to make the evaluation session transparent.

7.2.3. The medical and computational qualifications needed for specification

First, it was found that the collaboration of an EP and a KE is crucial for successful formal specification of a GL. In particular, as previously emphasized, creating an OSC is an indispensable, crucial step before markup which should be performed by a senior (experienced) EP and a KE. It is suggested that senior EPs and KEs together should work on the tasks of selecting a GL for specification and making the GS, since they are familiar with the clinical and semantic aspects. However, once an OSC and a textual representation of a GL are provided, it was found that any EP (senior, non-senior or a general physician) can structure the GL's knowledge in a semiformal representation completely (see section 5.2 regarding comprehensive results). However, to specify it correctly, an EP with good computational skills should be selected, perhaps from among residents, interns or even students, who have less practical experience in the medical field, but who can perform complex computational tasks such as markup, and can use the specification tools (possibly due to specific training in the tools and the ontology).
Finally, it is suggested that the task of evaluating the markups using the Evaluation Markup Tool, should be done by a KE with a Senior EP, who can be a "referee" for measuring the completeness and correctness measures.

7.2.4. The characteristics of the KA tool needed for this kind of specification

The low usability of The URUZ tool (see result 7 in 5.1) was the trigger for understanding that a more robust, graphical, highly usable framework is needed, in order to make the structuring process easy and transparent, especially for EPs who have less computational orientation than a KE. Using the GESHER framework might increase the quality of the markup of KRs in the *Condition*s and *Plan-Body* KR classes. More intuitive, graphic, user friendly interfaces should be used in order to reduce to a minimum the need for the EP to know the Asbru semantics of complex KRs (such as *Cyclical*, *Sub-Plans* or *Abort Condition*), and to bridge the gap between the initial structuring of the EP and the full semantics of the specification language.
The completeness of the markup of declarative KRs, such as the KRs in the *Conditions* KR class, seems to be lower; therefore, interfaces targeting these KRs, such as The Expression Builder that was developed within the GESHER framework (see section 6.3.2) might be used to increased the completeness level for those KRs. In addition, as mentioned before, GESHER might be used for the creation and specification of the OSC, enable creation and re-using of graphical procedural patterns which can be sharable among different OSC.

7.3. Limitations and Advantages of the research

An apparent limitation of this study was the small of the number of EPs and GLs, which is a common limitation in KA evaluations. However, in fact, 196 sub-plans and 326 KRs in total were structured by all of the EPs together in all markups, enabling us to capture in detail all the specification phases from free-text representation into semi-formal representation. Another potential limitation was the lack of careful measurement of the required time: since the EPs worked mainly in their spare time (which was limited anyhow), it was hard to capture the influence of the time variable on this research, and to measure, for example, the precise time it took an EP to structure a GL or each KR. However, not working under a rough time constraints, enabled us to obtain more realistic results, since the interaction with most of the EPs took place in their own "playground". That is, the editing was often performed within their clinical settings, in their spare time, between their "real" tasks of treating patients, a situation that is as close as possible to our objective of examining the option of specification of GLs by EPs who are themselves involved in the point of care, and not in an "artificial" environment of a lab.

7.4. Final Words

Although it seems that the creation of an OSC is time consuming and creates a "bottle neck" in the specification process, the opposite might be true; that is, it might save time and costs: instead of a programmer creating and maintaining a version of a GL, which was not necessarily created in collaboration with an EP, here a process is suggested in which GLs are specified as soon as possible in the process as OSCs, with an agreement among all professional EPs and KEs. Thus, the specification process itself can be performed by less professionally advanced physicians, with high computational skills, for example, medical students or interns, who are more accessible than a Senior EP and a KE working together, performing the whole process.
Future research should include evaluation of a graphical framework such as GESHER, with time measurement of all tasks. In addition, the specification process

could be performed by medical oriented students using multiple versions of OSCs for the same GL, each OSC in a different specification language and with a different level of detail, trying to find the ideal structure and granularity level of an OSC.
An Application of the markups by runtime application tools such as SPOCK might be suggestion for future research too, maybe even use these application tools within the process of making OSC as a "debugger" tool for finding problems in the procedural (and declarative) structure of the OSC, before release it to the editors for structuring.
Finally, using an explicit detailed OSC, highlights the importance of the need of structuring GL from the beginning in a way that is not ambiguous and that is geared as much towards appropriate ontology. It might be good for GL designers to work from a certain stage with a tool such as GESHER, creating an OSC as base for further GL markup structuring.

8. References

Bennett, J. S. (1985). "ROGET: A knowledge-based system for acquiring the conceptual structure of a diagnostic expert system." Journal of Automated Reasoning, **1**: 49-74.

Boaz, D. and Shahar Y. (2005). "A framework for distributed mediation of temporal-abstraction queries to clinical databases." Artificial Intelligence in Medicine **34(1)**: 3-24.

Boose, J. H. (1985). "A knowledge acquisition program for expert systems based on personal construct psychology." International Journal of Man-Machines Studies **23**: 49.5-525

Boxwala, A., Peleg M. and Tu S. e. a. (2004). "GLIF3: a representation format for sharable computer-interpretable clinical practice guidelines." J Biomed Inform **jun;37(3)**: 147-61.

Brooke, J. (1996). "SUS - A quick and dirty usability scale." http//:www.usability.serco.com/trump/documents/Suschapt.doc.

Ciccarese, P., Caffi E., Boiocchi L., Quaglini S. and Stefanelli M. (2004). A guideline management system. Medinfo.

Coiera, E. (2000). "When conversation is better than computation." JAMIA **7,Number 3**.

COPD (2005). "Clinical Practice guidelines, VA Hospital http://www.oqp.med.va.gov/cpg/COPD/copd_cpg/content/b1/annoB1.htm."

De Clercq, P., Blom J., Korsten H. and Hasman A. (2004). "Approaches for creating computer-interpretable guidelines that facilitate decision support." Artif Intell Med **31(1)**: 1-27.

De Clercq, P. and Hasman A. (2004). Experiences with the Development, Implementation and Evaluation of Automated Decision Support Systems. Medinfo.

Dexter, P. R., Perkins S., Overhage J. M., Maharry K., Kohler R. B. and McDonald C. J. (2001). "A Computerized Reminder System to Increase the Use of Preventive Care for Hospitalized Patients." The New England Journal of Medicine **345**: 965-970.

Elkin, P., Peleg M., Lacson R., Bernstam E., Tu S., Boxwala A. and Greenes R. (2000). "Toward the standardization of electronic guidelines." MD Comput **17(6)**: 39-44.

Fedson, D. S. (1994). "Adult Immunization: summary of the national vaccine advisory committee report." JAMA **272**: 1133-7.

Feigenbaum, E. A. (1984). "Knowledge engineering: The applied side of artificial intelligence." Annals of the New-York academy of sciences **246**: 91-107.

Fenton, N. E. (1991). Software Metrics. London, London: Chapman and Hall.

Field, M., Lohr K. and Eds (1990). "Clinical practice guidelines :Directions for a new program." National Academy Press.

Fox, J., Johns N. and Rahmanzadeh A. (1998). "Disseminating medical Knowledge: the PROforma approach." Artificial Intelligence in Medicine **14**: 157-181.

German, E. and Shahar Y. (2005). MEIDA: A Generic Architecture for Linking Medical Decision-Support Applications to Clinical Databases. Medical Informatics research Cebter, Department of Information Systems Engineering, Ben Gurion University, Beer Sheva, Israel.

Goldstein, M. K., Hoffman B. B., Coleman R. W., Tu S. W., Shankar R. D., O'Connor M., Martins S., Advani A. and Musen M. A. (2001). Patient Safety in Guideline-Based Decision Support for Hypertension Management: ATHENA DSS. AMIA Annual Symposium, Washington, DC.

Gordon, C. and Veloso M. e. a .(1999) .Guidelines in Healthcare: the experience of the Prestige project. Proceedings of MIE'99.

Grimshaw, J. M. and Russel I. T. (1993). "Effect of clinical guidelines on medical practice: A systematic review of rigorous evaluations." Lancet **342**: 1317-22.

Gruber, T. R. (1993). "A translation approach to portable ontologies." Knowledge Acquisition **5(2)**: 199-220.

Hagerty, C., Pickens D., Kulikowski C. and F. S. (2000). HGML: a hypertext guideline markup language. Proc AMIA Symp.

Herbert, S. I., Gordon C. J ,.Jackson-Smale A. and Renaud Salis J.-L. (1995). "Protocols for clinical care." Computer Methods and Programs in Biomedicine **48**: 21-26.

HL7-RIM http://www.hl7.org/Library/data-model/RIM/modelpage_mem.htm.

Hripcsak, G., Ludemann P., Pryor T., Wigertz O .and D C. (1994). "Rationale for the Arden Syntax." Computers and Biomedical Research **27**: 291-324.

HYPERTHYROIDISM-AACE (2002). "http://www.aace.com/pub/pdf/guidelines/hypo_hyper.pdf."

Johnson, P., Tu S. and N. J. (2001). Achieving reuse of computable guideline systems. Medinfo.

Johnson, P. D., .S.W. T., .N. B., .B. S. and .I.N. P. (2000). Using scenarios in chronic disease management guidelines for primary care In Overhage M.J., Ed. AMIA Annual Symposium, Los Angeles, CA.

Kaiser, K. (2005). Semi-automatic Transformation of Structured Guideline Components into Formal Process Representations. 1st Doctoral Consortium at the 10th Conference on Artificial Intelligence in Medicine (AIME 2005), Aberdeen, UK.

Karras, B., SD N. and Shiffman R. (2000). A Preliminary Evaluation of Guideline Content Mark-up Using GEM-An XML Guideline Elements Model. AMIA Annual Symposium, Los Angeles, CA.

Kosara, R. and Miksch S. M. o. M. A. V. a. U. I. f. T.-O., Skeletal Plans, (2001). "Metaphors of Movement: A Visualization and User Interface for Time-Oriented, Skeletal Plans." Artificial Intelligence in Medicine **22(2)**: 111-131.

Micieli, G., Cavallini A. and S Q. (2002). "Guideline Compliance Improves Stroke Outcome - A Preliminary Study in 4 Districts in the Italian Region of Lombardia." Stroke **33**: 1341-1347.

Miksch, S., Kosara R., Shahar Y. and Johnson P. (1998). AsbruView: Visualization of time-oriented, skeletal plans. In: The Fourth International Conference on Artificial Intelligence Planning Systems 1998 (AIPS-98), Carnegie-Mellon University, Pittsburgh, Pennsylvania.

Miksch, S., Shahar Y. and Johnson P. (1997). Asbru: A task-specific, intention-based, and time-oriented language for representing skeletal plans. Proceedings of the Seventh Workshop on Knowledge Engineering Methods and Languages (KEML-97), Milton Keynes, UK.

Moskovitch, R., Hessing A. and Shahar Y. V. (2004). Vaidurya - A concept-based, context-sensitive search engine for clinical guidelines. Proceedings of MEDINFO-2004, the Eleventh World Congress on Medical Informatics.

Musen, M. (1993). An overview of Knowledge Acquisition, in

J.M. David, J.P. Krivine, and R. Simmons (Eds). Second Generation Expert Systems, Berlin, Springer Verlag.

Musen, M., Tu S. W., Das A. and Shahar Y. (1996). "EON: A component-based approach to automation of protocol-directed therapy." Journal of the American Medical Information Association, 1996. **3(6)**: 367-388.

Musen, M. A., Carlson R. W., Fagan L. M. and Deresinski S. C. (1992). T-HELPER: Automated Support for Community-Based Clinical Research. Proceedings of the Sixteenth Annual Symposium on Computer Applications in Medical Care, Washington, D.C.

Musen, M. A., Fagan L. M., Combs D. M. and Shortliffe E. H. (1987). "Use of a domain model to drive an interactive knowledge-editing tool ".International Journal of Man-Medicine Studies **26**: 105-121.

Noy, N., Crubezy M. and Fergerson R. e. a. (2003). Protege-2000: An Open-source Ontology-development and Knowledge-acquisition Environment. Proc AMIA Symp.

Ohno-Machado, L., Gennari J. H. and al .M. S. N. e. (1998). "The guideline interchange format: a model for representing guidelines." Journal of the American Medical Informatics Association **5**: 357-72.

OpenClinical (2005). "http://www.openclinical.org."

Patel, V., Allen V., Arocha J. and Shortliffe E. (1997). "Representing Clinical Guidelines in GLIF: Individual and Collaborative Expertise." J Am Med Inform Assoc.. **5(5)**: 467-83.

Patel, V. L., Branch T., Wang D., Peleg M. and Boxwala A. A. (2002). "Analysis of the Process of Encoding Guidelines :An Evaluation of GLIF3." Methods of Information in Medicine **41(2)**: 102-113.

Peleg, M., Boxwala A. and al. T. S. e. (2004). "The InterMed approach to sharable computer-interpretable guidelines: a review." J Am Med Inform Assoc.. **11(1)**: 1-10.

Peleg, M., Boxwala A., Bernstam E., Tu S. W., Greenes R. and Shortliffe E. (2001). "Sharable Representation of Clinical Guidelines in GLIF: Relationship to the Arden Syntax." Journal of Biomedical informatics **34**: 170-181.

Peleg, M., Boxwala A. A. and al. O. O. e. (2000 .(GLIF3: The Evolution of a Guideline Representation Format. Proc. AMIA Annual Symposium.

Peleg, M., Gutnik L., Snow V. and .V.L. P. (2005). "Interpreting procedures from descriptive guidelines." Journal of Biomedical Informatics: in press.

Peleg, M., Tu S. W., Bury J., Ciccarese P., Fox J., Greenes R. A., Hall R., Johnson P. D., Jones N., Kumar A., Miksch S., Quaglini S., Seyfang A., Shortliffe E. H. and Stefanelli M. (2002). "Comparing Computer-Interpretable Guideline Models: A Case-Study Approach." JAMIA **10(1)**: 52–68.

PID-CDC (2002). "Centers for Disease Control and Prevention." http://www.cdc.gov/mmwr/preview/mmwrhtml/rr5106a1.htm.

PID-Emedicine (2005). "PID -http://www.emedicine.com/med/topic1774.htm."

Pryor, T. and Hripcsak G. (1993). Sharing MLM's :an experiment between Columbia-Presbyterian and LDS Hospital. Proceedings of the Seventeenth Annual Symposium on Computer Applications in Medical Care.

Quaglini, S., Stefanelli M., Lanzola G., Caporusso V. and Panzarasa S. (2001). "Flexible Guideline-based Patient Careflow Systems." Artif Intell Med **22**: 65-80.

Qualigini, S., Ciccarese P., Micieli G. and Cavallini A. (2004). Non-Compliance with Guidelines: Motivations and Consequences in a case study. Proccedings of the symposium on Computerized Guidelines s and Protocols (CGP 2004), Prague.

Ram, P., Berg D. and Tu S. e. a. (2004). Executing Clinical Practice Guidelines using the SAGE Execution Engine. Medinfo.

Ruzicka, M. and Svatek V. (2004). Mark-up based analysis of narrative guidelines with the Stepper tool. Proc. (CGP-04), Praha.

Seyfang, S., Kosara R. and S. M. (2000). Asbru's Reference Manual, Asbru Version 7.2, Document Revision 0., Technical Report, Asgaard-TR-2000-3., The Vienna Asgaard Project, Vienna University of Technology, Institute of Software Technology, Vienna.

Shadbolt, N., O'Hara K. and Crow L. (1999). "The Experimental Evaluation of Knowledge Acquisition Techniques and Methods: History, Problems and New Directions." International Journal of Human-Computer Studies, special issue on evaulation of KA techniques **51(4)**: 729-755.

Shahar, Y., Miksch S. and 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**: 29-51.

Shahar, Y., Shalom E., Mayaffit A., Young O., Galperin M., Martins S. B. and Goldstein M. K. (2003). A distributed, collaborative, structuring model for a clinical-guideline digital-library. AMIA Annual Fall Symposium, Washington, DC.

Shahar, Y., Young O., Shalom E., Galpering M., Mayaffit A., Moskovitch R. and A H. (2004). "A Framework for a Distributed, Hybrid, Multiple-Ontology Clinical-Guideline Library and Automated Guideline-Support Tools." Journal of Biomedical Informatics **37(5)**: 325-344.

Shalom, E. and Shahar Y. (2005). A Graphical Framework for Specification of Clinical Guidelines at Multiple Representation Levels. AMIA Annual Fall Symposium, Washington DC, USA.

Shankar, R. D., Tu S. W. and Musen M. A. (2002). Use of Protege-2000 to Encode Clinical Guidelines. AMIA Annual Symposium.

Sherman, E., Hripcsak G., Starren J., Jenders R. and Clayton P. (1995 .(Using Intermediate States to Improve the Ability of the Arden Syntax to Implement Care Plans and Reuse Knowledge. Proceedings of the Annual Symposium on Computer Applications in Medical Care (SCAMC-95), New Orleans, LA.

Shiffman, R., Karras B., Agrawal A., Chen R., Marenco L. and Nath S. (2000). "GEM: a proposal for a more comprehensive guideline document model using XML." Journal of the American Medical Informatics Association: 488-498.

Shiffman, R., Michel G., Essaihi A. and Thornquist E. (2004). "Bridging the guideline implementation gap: a systematic, document-centered approach to guideline implementation." J Am Med Inform Assoc.. **11(5)**: 418-26.

Shortliffe, E. H., Charles P. F., Jeremy C. W., Smith A. C. and Kaplan B. (2000). Evaluation methods in medical informatics, Springer.

Shortliffe, E. H., Scott A., Bischoff M., van Melle C. and Jacobs W. (1981). ONOCIN: An Expert system for Oncology Protocol Management. Proceedings of the 7th international Joint conference on artificial intelligence.

Siegel ,S. and Castellan N. j. (1988). NONPARAMETRIC STATISTICS, McGRAW-HILL International Editions.

Sordo, M., Ogunyemi O., Boxwala A., Greenes R. and Tu S. (2004). "GELLO: An Object-Oriented Query and Expression Language for Clinical Decision Support." Summary Report prepared for OpenClinical.

Sutton, D. R. and .J. F. (2003). "The Syntax and Semantics of the PROforma guideline modelling language." J Am Med Inform Assoc.. **10(5)**: 433-43s.

Svatek, V. and Ruzicka M. (2003). "Step-By-Step Mark-Up of Medical Guideline Documents,." International Journal of Medical Informatics **70**: 329-335.

Terenziani, P., Montani S., Bottrighi A., Torchio M. and Molino G. (2002). Supporting physicians in taking decisions in clinical guidelines: the GLARE "What if" facilty. AMIA.

Terenziani, P., Montani S., Bottrighi A., Torchio M., Molino G. and Correndo G. (2004). The GLARE approach to clinical guidelines: main features. Symposium on Computerized Guidelines and Protocols (CGP) 2004, Prague.

Tu, S., Musen M. and al. S. R. e. (2004 .(Modeling guidelines for integration into clinical workflow. Medinfo.

Tu, S. W., Kahn M. G., Musen M. A., Ferguson J. C., Shortliffe E. H. and Fagan L. M. (1989). "Episodic Skeletal-plan refinement on temporal data." Communications of ACM **32**(1439-1455.(

Tu ,S. W. and Musen M. A. (1999). A flexible approach to guideline modeling. Proc AMIA Symp.

Van Bemmel, J. H. and Musen M. A. (1997). Strategies for Medical Knowledge Acquisition, the HandBook of medical informatics, Heidelberg:Springer.

Votruba, P., Miksch S. and R. K. (2004). Facilitating Knowledge Maintenance of Clinical Guidelines and Protocols. 11th World Congress of Medical Informatics (MedInfo 2004), AMIA.

Votruba, P. S. K. A. f. A. M. s. T., Institute of Software Technology and Interactive Systems ,Vienna University, Vienna, Austria, 2003 (2003). Structured Knowledge Acquisition for Asbru. Master's Thesis, Institute of Software Technology and Interactive Systems, Vienna University, Vienna,Austria.

W3C (2005). "http://www.w3.org./"

Walpole, R. E. and H.Myers R. (1978). Probability and Statistics for Engineers and Scientists, Macmillan publishing Co., Inc. New York

Collier Macmillan Publishers London.

Wang, D., Peleg M., Bu D., Cantor M., Landesberg G., Lunenfeld E., Tu S., Kaiser G., Hripcsak G., Patel V. and Shortliffe E. (2003). "GESDOR – a generic execution model for sharing of computer-interpretable clinical practice guidelines." Proc AMIA Symp.

Wang, D., Peleg M., Tu S., Boxwala A., Greenes R., Patel V. and Shortliffe E. (2002). "Representation primitives, process models and patient data in computer-interpretable clinical practice guidelines: a literature review of guideline representation models." Int J Med Inform **68(1-3)**: 59-70.

Wang, D., Peleg M. and Tu S. W. e. a. (2004). "Design and implementation of the GLIF3 guideline execution engine." Journal of Automated Reasoning, **Oct;37(5)**: 305-18.

Young, O. (2005). Runtime Application of Hybrid-Asbru Clinical Guidelines. Medical Informatics Research Center, Department of Information Systems Engineering, Ben Gurion University,Beer Sheva, Israel.

Appendixes

Appendix A – The Markup Kit

* Note: Originally, the markup kit contains the Ontology Specific Consensus(OSC) file, and the text of the guideline source. Those sections were removed in this markup kit because all the OSCs are in Appendix B. However the sources of the GLs, which were originally included too in this kit can be fount at :PID - . [PID-Emedicine 2005] , COPD - [COPD 2005] , HypoThyrodism - [HYPERTHYROIDISM-AACE 2002]

Mark Up Kit
For
URUZ Evaluation

DeGeL Team
Medical Informatics Research Center
Department of Information Systems Engineering

1. Steps for MarkUp in URUZ

1.1. Pre markup steps :

1.1.1. Read sources in advanced

1.1.2. Read consensus

1.1.3. train in order to have some experience with URUZ

1.2. markup steps

1.2.1. preparation for markup

1.2.1.1. Enter DeGeL system - http://medinfo1.ise.bgu.ac.il/DeGeL2

1.2.1.2. Select your group (BGU)

1.2.1.3. On the left menu choose "URUZ" for opening URUZ in order to create document for markup. "

1.2.1.4. open new guideline document See No 1 in "How To " table, give it a name and language(Hybrid-Asbru).In the case that more then one source, See No 2

1.2.1.5. if you have an already marked-up version, you can search it and continue working on it - See No 4

1.2.2. choose semantic indices

1.2.3. perform markup according to sections2,3,4,5

1.3. Post markup - **Checklist**

1.3.1. the markup is according to the consensus

1.3.2. all plans and sub plans were defined

1.3.3. all defined plans have text content

1.3.4. all "To-Be-defined" are for pointing to external plans

2. "How to.." in URUZ

No.	How To	Description
1	Open New Document	in the upper menu "New" -> "New document from guideline Doc search
2	Add Source	1. in the upper menu "Sources"->"add new source document" 2. search and select the guideline (see No.3)
3	Search for Source (with the "title")	1. Browse in the tree "Guideline Source Documentation" -> "Identity" -> "title" 2. add your key -word. 3. click on the "add" button 4. click on the "search" button 5. select the guideline source with the checkbox 6. click on "return result" button
4	Search for document(with the "title")	1. in the upper menu "Search"->"search for Mark-up Guideline document" 2. In the Left Lower tree browse to-> "Documentation"->"title" 3. add your key -word. 4. click on the "add" button 5. click on the "search" button 6. select the guideline source with the checkbox 7. click on "return result" button

3. MarkUp knowledge roles in URUZ

3.1. MarkUp knowledge roles common to all specification languages

(the nodes in the upper left tree excluding "Guideline Marked-Up Content" Node)

Knowledge Role name	Description	Example	Points to Keep in mind while markup
Semantic indices	The guideline classification according to the library's semantic axes.	Open the IndexiGuide Tool in the upper menu and select a node form the axes on the left. Save your work with "save" button and close the window with "close"	• Has semantic classification according to at least one axis • Better scoring for classification according to **multiple** axis (for more then one axis)
Guideline knowledge	• Contains definitions of medical concepts which might be used in several plans such as : Clinical parameters , their definitions , and their classification criteria. • any complex criteria like filter or setup conditions) which combines Or/And operators and	• Complex time definitions (e.g., the two weeks of moderate anemia), • classification tables of clinical parameters like HBG State • risk group specification • concept name is **filter condition for COPD** with this specification : (underlying COPD) **AND** (acute exacerbation COPD**-defined in guideline knowledge,** presence of any of items)	• Has detailed declarative knowledge about most clinical concepts in the guideline.
Level of evidence	Which grades of evidenced based researches this current plan can be based on?	**See Appendix A in page 9** for detailed scaling of Level of evidence	• Has detailed Level of evidence content base on the guideline source (if there is any). • Better scoring might be for define level of evidence for each sub plan
Strength of recommendation	What is the level we of recommendation of this plan	**See Appendix B in page 9** for detailed scaling of Strength of recommendation	• Has detailed Strength of recommendation content base on the guideline (if there is any). • Better scoring might be for define of **Strength** recommendation for each sub plan

Actors	Specifies who is responsible or taking part in performing the guideline actions	Nurse, gynecologist, etc.	• Has detailed Actors content. • Better scoring might be for defining some actors and not just one (e.g. nurse and physician) based guideline directives • Better scoring might be for define Actors for each sub plan
Clinical context	Specifies where in the clinical setting the patient is being seen	outpatient clinic, ER, ICU, extended care, etc.	• Has detailed clinical context base on guideline directive

3.2. MarkUp knowledge roles of Asbru Ontology

(under "Guideline Marked-Up Content" Node)

Knowledge Role name		Description	Example	Points to Keep in mind while markup
Intentions (goals)	*Intermediate process*	The **action(s)** that should take place during the **process of the plan**	monitor blood glucose once a day	• the current content is according to *Intermediate process* semantics • the necessary knowledge is defined in the guideline knowledge
	overall process	The **action(s)** that should take place **after finishing the plan**	patient had visited dietitian regularly for at least three month	• the current content is according to *overall process* semantics • the necessary knowledge is defined in the guideline knowledge
	Intermediate outcome	The **state(s)** that should be achieved, or maintained, or avoided **during the process** of the plan	• avoid complications auto PEEP • avoid acute respiratory alkalosis and	• the current content is according to *Intermediate outcome* semantics • the necessary knowledge is defined in the guideline knowledge
	overall outcome	The **state(s)** that should achieved, or maintained, or avoided **after finishing the plan**	patient had less than one high glucose value per week	• the current content is according to *overall outcome* semantics • the necessary knowledge is defined in the guideline knowledge

Conditions	*Filter condition*	• Specifies the exclusion/inclusion criteria of the guideline • When this criteria is true ,the current plan becomes possible to apply (see section 5) • This criteria must hold during the entire selection phase • if it doesn't hold, unlike the setup precondition, it cannot be checked again.	1.((Patient must be female) **AND** (underlying COPD) **AND** ((acute exacerbation COPD-defined in guideline knowledge)) **OR** Pulmonary hypertension))	• When it is complex, there should be And/Or operators between the different criteria • the current content is according to *Filter condition* semantics • the necessary knowledge is defined in the guideline knowledge
	setup	• Specifies the additional criteria which should be achieved through actions by the physician prior to the start of plan applying • This actions can have time limit to be achieved • When this criteria is true ,the current plan **selected** and ready for Appling (see section 5) and then **activated**	• Patient need to have a positive glucose-tolerance test **AND** hgbA_1C > 4.5 • **Allow a week** for pregnancy test • **Wait for a day** to check whether patient temperature greater than 39°C	• When it is complex, there should be And/Or operators between the different criteria • the current content is according to setup condition semantics • the necessary knowledge is defined in the guideline knowledge • A better scoring is for define a time limit for this criteria to be achieved
	abort	• Specifies when a plan must end unsuccessfully • When this criteria is true ,the current plan is **aborted** and ends (see section 5) • It is possible to add specification of what should be done when plan abort	• There is an indication for insulin treatment: the patient cannot be controlled by diet **OR** hgbA_1C > 4.5 • patient death • if patient state becomes acute(the abort condition), them start a surgery procedure	• When it is complex, there should be And/Or operators between the different criteria • the current content is according to *abort* condition semantics • the necessary knowledge is defined in the guideline knowledge

Conditions(cont)	*Suspend*	• Specifies when a plan must be put on a hold • When this criteria is true, the current plan is **suspended** (see section 5) and in hold until aborted or reactivate • It is possible to add specification of what should be done when plan abort	• Patient's blood glucose has been high for at least four days • If Patient has high normal hypertension for 2 weeks (the suspend condition), start diet procedure	• When it is complex, there should be And/Or operators between the different criteria • the current content is according to *Suspend* condition semantics • the necessary knowledge is defined in the guideline knowledge
	reactivate	• Specifies when a plan can be reactivated after being suspended • When this criteria is true ,the current plan becomes **activated** again (see section 5)	• Patient's blood glucose has been normal for at least one day • Patient has normal hypertension (**concept defined in guideline knowledge**) for at least 2 weeks	• When it is complex, there should be And/Or operators between the different criteria • the current content is according to *reactivate* condition semantics • the necessary knowledge is defined in the guideline knowledge
	complete	• Specifies when a plan can end successfully • When this criteria is true ,the current plan becomes **complete** (see section 5)	• Delivery has been performed • patient has no symptoms of COPD • patient discharged	• When it is complex, there should be And/Or operators between the different criteria • the current content is according to *complete* condition semantics • the necessary knowledge is defined in the guideline knowledge

3.3. Working with Plan Body Wizard (PBW):

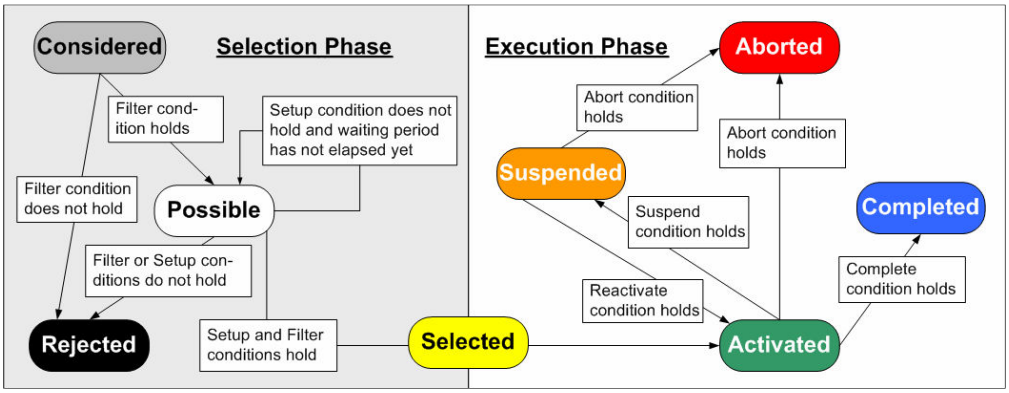
Knowledge Role name		Question in the wizard	Description	example	Points to Keep in mind while markup
plan body	*simple action*	Is the current step a simple action?	An atomic plan with simple semantics. Suitable for defining plans with one action	• Give prescription • Check lab test value • Measure patient temperature	• Has a text content describing the plan • Has single atomic action semantics with clear specification and description for the action to be performed • the necessary knowledge is defined in the guideline knowledge
	Pre-defined plan	Does the current step refer to a pre-defined plan in the DEGEL library?	Used to refer a plan which was defined in DeGeL	• Every plan which was defined in DeGeL and can be reused	• Has a text content describing the plan • DeGeL ID is defined
	Repeating Plan	Is the current step a periodic action (repeated twice or more)? (The third option)	A plan that should be repeat more then on time in periods	• BP measure once a day between 1 to 3 weeks • Take drug A three times a day for 3 months • Perform clinical assessments every day after treatment • If patient improving take drug every hour, else take drug every 3	Has a text content describing the plan • Has a semantic of repainting, starting Specification and frequency duration • Better scoring for plans with complete time specification • Has repeating semantics • the necessary knowledge is defined in the guideline knowledge

Plan body (cont)	*Repeating plan (cont)*	Repeated action or Plan Label		The name of the action	• BP measure • Take drug	This plan should be a defined also with the "Plan Body Wizard"
		Starting Specification		**When to start** to apply the plan for the first time	• Now • Start of the treatment plan	The default of this property is "Now", which gets a date and time measures when applying by Spock
		Describe frequency duration and specific conditions for the activation		What is the **frequency or the conditional frequency** of the plan	• once a day • three times a day • If patient improving take drug every hour, else take drug every 3	Note that the frequency can be simple , like once a day or conditional frequency like If patient improving take drug every hour, else take drug every 3
		The specification about the complete time or condition of the overall periodic plan	Specification about the end time of the overall periodic plan	stop the plan **after it passes this end time**	• After 3 months • After 2 weeks	
			conditions for the overall periodic plan completion:	stop the plan **when it meets this criteria**	• Patient improving • Needs hospitalization	
			Number of required successful completions	stop the plan **after those number of repetitions**	3,4…	

	Sub plan	Is the current step composed of **two or more steps** ?	**Parallel** Order There are two or more sub plans which possibly overlap – (Parallel)	• plan which is composed from more then one sub plans. Note that the order between those sub plans should be defined two (like parallel order)	• Plan which is composed the sequential steps diagnosis, treatment and follow-up • Plan for selecting regimens which composed of three different regimens with no particular order • Plan for give 2 drugs in parallel	Has a text content describing the plan • defining sub plan content for each plan as well (All sub -plans are further defined) • At least 2 sub-plans must be defined • order must be defined between plans (parallel, sequential or unordered) • the necessary knowledge is defined • There is specification for all "hard" elements: mandatory, max attends, waiting strategy, completed spec..
			Sequential Order At each moment in time only one plan is performed ->.(Sequential)			
		Finishing specification	• Which subplans must be completed ?	• How many subplans must complete successfully from all the subplans?	• One • 1..15 • some • none	• this specification comes instead of the "mandatory" column. In other words you can do one of the follwing but not bothe of them : specify which plans are mandatory Or specify how many must complete
			• Attempt to finish all optional subplans according to the parent plan time limit ?	• In case that there are optional subplans: do you want to try to finish them (apply them) , despite they don't have to be finish?	• Yes • no	•
			• Try each subplan more than once if fails ?	• If some plan is failed from any reason, do you want to try to apply it again?	• Yes • no	•

	if-then-else	• Does the current step have a condition e.g., if - then? -**>next** • Does the condition have a single possibility, that it if holds, perform a specified action\s or otherwise perform a different action\s ?	• A condition between 2 plans. When this condition holds ("yes") , then do plan A. when this condition is false do plan B	• Is mechanical ventilation required? If yes, then start mechanical ventilation, else start Oxygen therapy • Is patient improving?. If yes, consider discharge, else continue in hospitalization.	Has a text content describing the plan • Has if-then-else semantics • Has clear specification for the Condition • Has definition to "then" case • Has definition to "else" case when it says there should be one. • All alternative steps are further defined • the necessary knowledge is defined
	switch case	• Does the current step have a condition (e.g., if - then)? -**>next** • Is there a criteria to check and different actions to perform for different values of the criteria ?	• The criteria has some possible values. For each value, a plan should be defined	• If hypertension state is normal do follow-up, if it is mild do treatment if it severe hospitalize patient. • **inspected value**: hypertension state **Possible values** : - normal -> follow-up mild -> treatment severe-> hospitalize	Has a text content describing the plan • Has switch case semantics • The criteria has multiple values • All alternative steps are further defined • Has at least 2 Possibility Values • the necessary knowledge is defined in the guideline knowledge
	To-Be-defined	• Other	• This plan is not in the scope of this guideline, and needed to be define later in DeGeL as separate GL Document	• Defining Emergency operation plan within some treatment	

3.4. The states a plan can be in during its application



3.5. Appendix A - Scales for Level Of Evidence :

LEVELS OF EVIDENCE	
1++	High quality meta-analyses, systematic reviews of RCTs, or RCTs with a very low risk of bias.
1+	Well conducted meta-analyses, systematic reviews of RCTs, or RCTs with a low risk of bias.
1-	Well conducted meta-analyses, systematic reviews of RCTs, or RCTs with a high risk of bias.
2++	High quality systematic reviews of case-control or cohort studies with a very low risk of confounding, bias, or chance and a high probability that the relationship is causal.
2+	Well conducted case control or cohort studies with a low risk of confounding, bias, or chance and a moderate probability that the relationship is causal.
2-	Case control or cohort studies with a high risk of confounding or bias and a signifi cant risk that the relationship is not causal.
3	Non-analytic studies, eg, case reports, case series.
4	Expert opinion.

3.6. Appendix B - Scales of strength of Recommendations :

GRADES OF RECOMMENDATION	
A	At least one meta-analysis, systematic review, or RCT rated as 1++, and directly applicable to the target population; **OR** A systematic review of RCTs or a body of evidence consisting principally of studies rated as 1+, directly applicable to the target population, and demonstrating overall consistency of results.
B	A body of evidence including studies rated as 2++, directly applicable to the target population, and demonstrating overall consistency of results; **OR** Extrapolated evidence from studies rated as 1++ or 1+.
C	A body of evidence including studies rated as 2 +, directly applicable to the target population and demonstrating overall consistency of results; **OR** Extrapolated evidence from studies rated as 2++.
D	Evidence level 3 or 4; **OR** Extrapolated evidence from studies rated as 2+.

Appendix B – Ontology Specific Consensus files

Appendix B.1 - The in-formal consensus of the COPD GL, (first version)

Actors: point of care physician ED and wards

Clinical Context: ED and ward

Conditions

Entry conditions:

Filter conditions: (underlying COPD) AND (acute exacerbation COPD-defined in guideline knowledge, presence of any of items)

Setup conditions: none

Abort conditions: does not meet advance directives (patient's wishes)

Complete conditions: discharge, death

Suspend conditions: CPR

Restart conditions: (CPR resolved and COPD acute exacerbation present)

Intentions

Process Intentions:

Intermediate Process Intentions:

1.Decrease frequency of inhaled $beta_2$-agonists to every 4 to 6 hours
2.Switch to MDI with spacers.
3.Switch from parenteral to oral medication.
4.Titrate oxygen as per oxygen protocol

Overall Process Intentions:

Outcome Intentions:

Intermediate Outcome Intentions: avoid complications auto PEEP, avoid acute respiratory alkalosis and

Improvement is indicated by:
Reduced dyspnea.
Decreased respiratory rate.
Improved air movement.
Decreased use of accessory muscles.
Improved peak expiratory flow.
Improved FEV_1 and/or ABGs.

Overall Outcome Intentions:

Features of the severe exacerbation are resolved (see Annotation D).
Anticipated need for inhaled bronchodilators is not more frequent than every 4 hours and the patient is on oral medication.

Reversible component of airway obstruction, if present, is under stable control.

Patient or caregiver understands appropriate use of medications.

Follow-up and home care arrangements have been completed (e.g., visiting nurse, oxygen delivery, meal provisions).

Patient, family, and physicians are confident that the patient can manage successfully.

Guideline knowledge: sections A,B,D,E,I and G

Level of Evidence

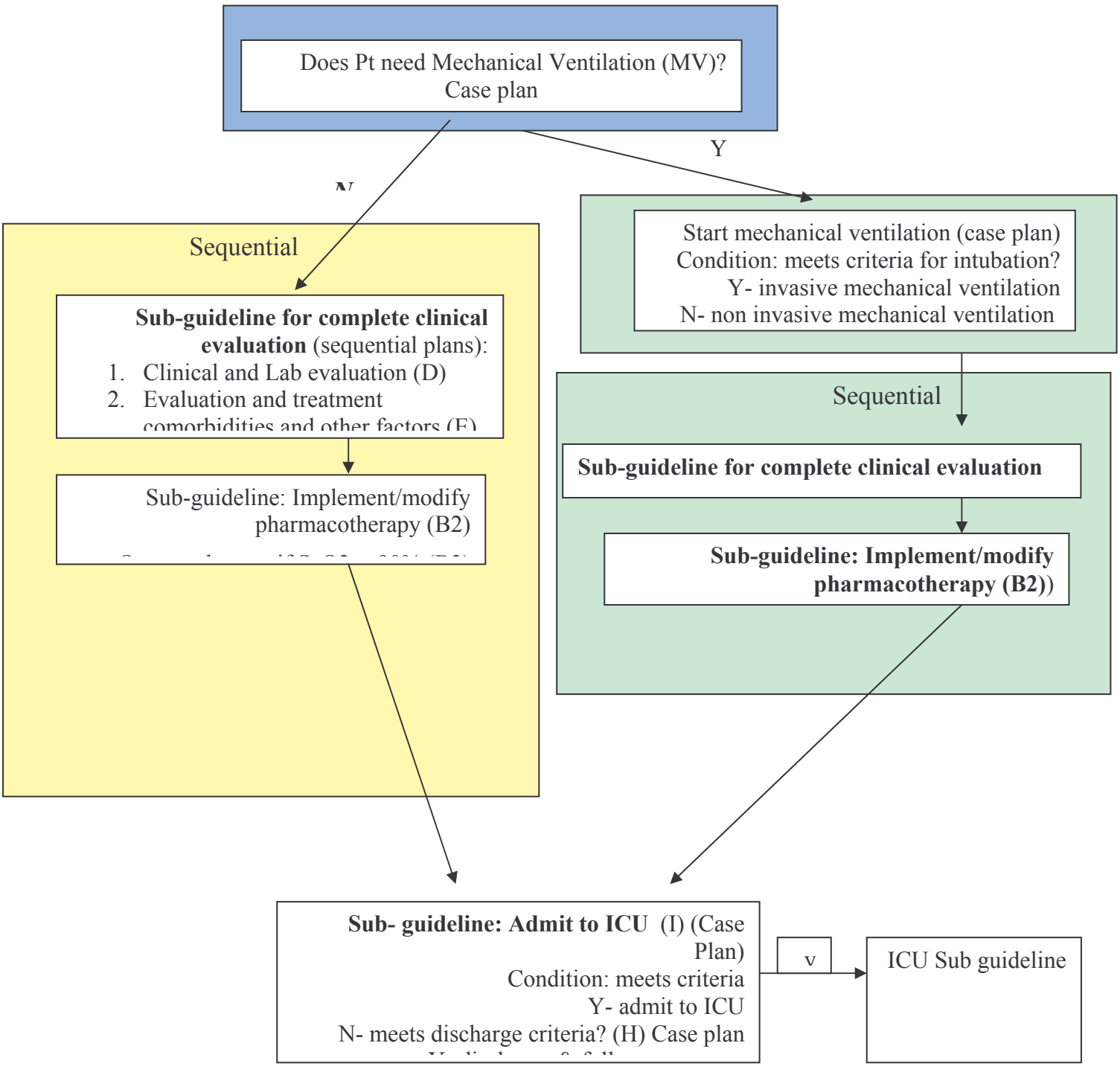
Strength of recommendation

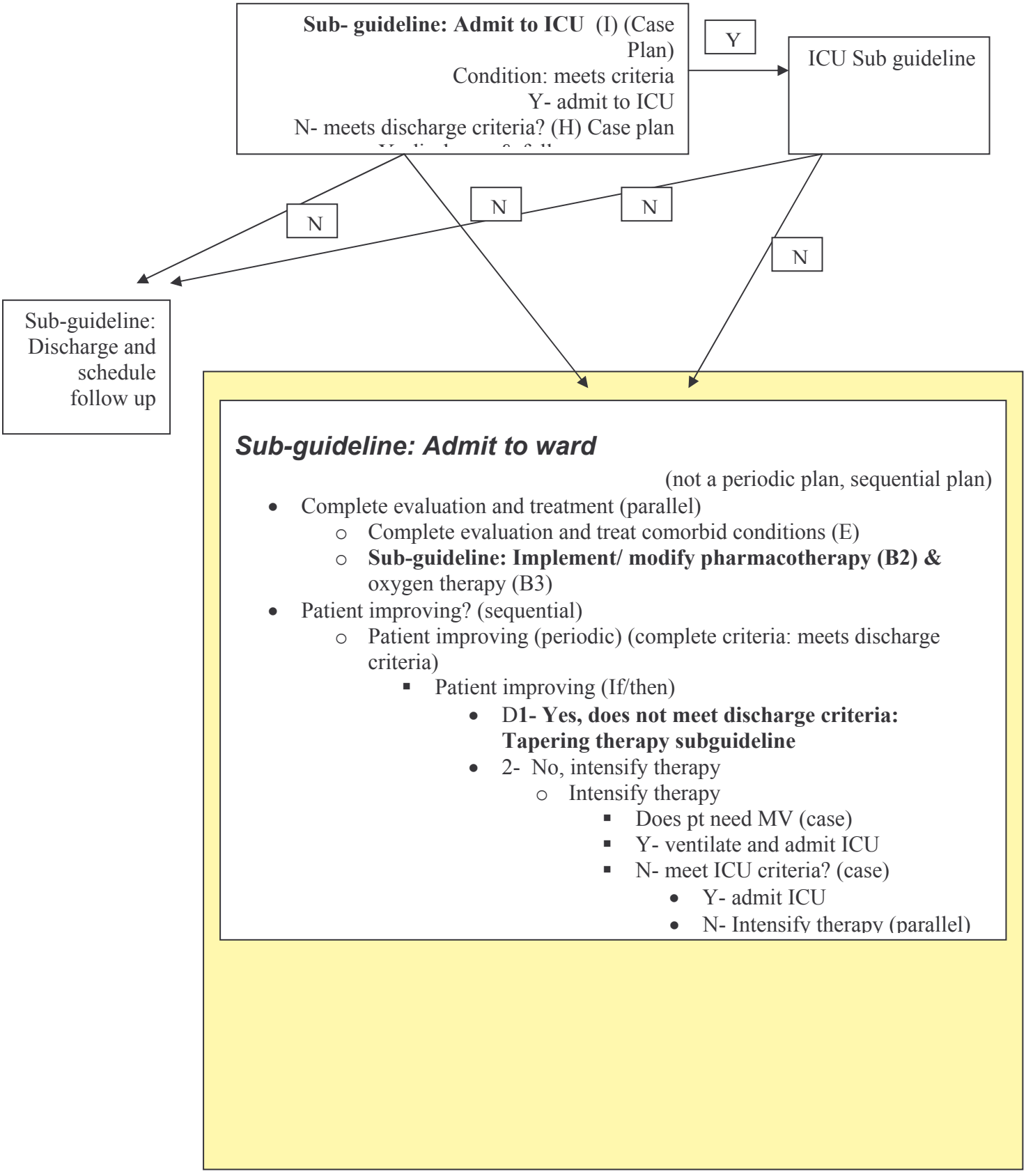
Planbody

Preferences

Effects: Discharge from ward

PBW





Appendix B.2 - The textual source of the "Inpatient Treatment" plan as taken form the CDC

No efficacy data compare parenteral with oral regimens. Many randomized trials have demonstrated the efficacy of both parenteral and oral regimens (*82*). Although most trials have used parenteral treatment for at least 48 hours after the patient demonstrates substantial clinical improvement, this time designation is arbitrary. Clinical experience should guide decisions regarding transition to oral therapy, which usually can be initiated within 24 hours of clinical improvement. Most clinicians recommend at least 24 hours of direct inpatient observation for patients who have tubo-ovarian abscesses, after which time home antimicrobial therapy is adequate.

Parenteral Regimen A

Cefotetan 2 g IV every 12 hours

OR

Cefoxitin 2 g IV every 6 hours

PLUS

Doxycycline 100 mg orally or IV every 12 hours.

NOTE: Because of pain associated with infusion, doxycycline should be administered orally when possible, even when the patient is hospitalized. Both oral and IV administration of doxycycline provide similar bioavailability.

Parenteral therapy may be discontinued 24 hours after a patient improves clinically, and oral therapy with doxycycline (100 mg twice a day) should continue to complete 14 days of therapy. When tubo-ovarian abscess is present, many health-care providers use clindamycin or metronidazole with doxycycline for continued therapy rather than doxycycline alone, because it provides more effective anaerobic coverage.

Clinical data are limited regarding the use of other second- or third-generation cephalosporins (e.g., ceftizoxime, cefotaxime, and ceftriaxone), which also may be effective therapy for PID and may replace cefotetan or cefoxitin. However, these cephalosporins are less active than cefotetan or cefoxitin against anaerobic bacteria.

Parenteral Regimen B

Clindamycin 900 mg IV every 8 hours

PLUS

Gentamicin loading dose IV or IM (2 mg/kg of body weight) followed by a maintenance dose (1.5 mg/kg) every 8 hours. Single daily dosing may be substituted.

Although use of a single daily dose of gentamicin has not been evaluated for the treatment of PID, it is efficacious in other analogous situations. Parenteral therapy can be discontinued 24 hours after a patient improves clinically; continuing oral therapy should consist of doxycycline 100 mg orally twice a day or clindamycin 450 mg orally four times a day to complete a total of 14 days of therapy. When tubo-ovarian abscess is present, many health-care providers use clindamycin for continued therapy rather than doxycycline, because clindamycin provides more effective anaerobic coverage.

Alternative Parenteral Regimens

Limited data support the use of other parenteral regimens, but the following three regimens have been investigated in at least one clinical trial, and they have broad spectrum coverage.

Ofloxacin 400 mg IV every 12 hours

OR

Levofloxacin 500 mg IV once daily

WITH or WITHOUT

Metronidazole 500 mg IV every 8 hours

OR

Ampicillin/Sulbactam 3 g IV every 6 hours

PLUS

Doxycycline 100 mg orally or IV every 12 hours.

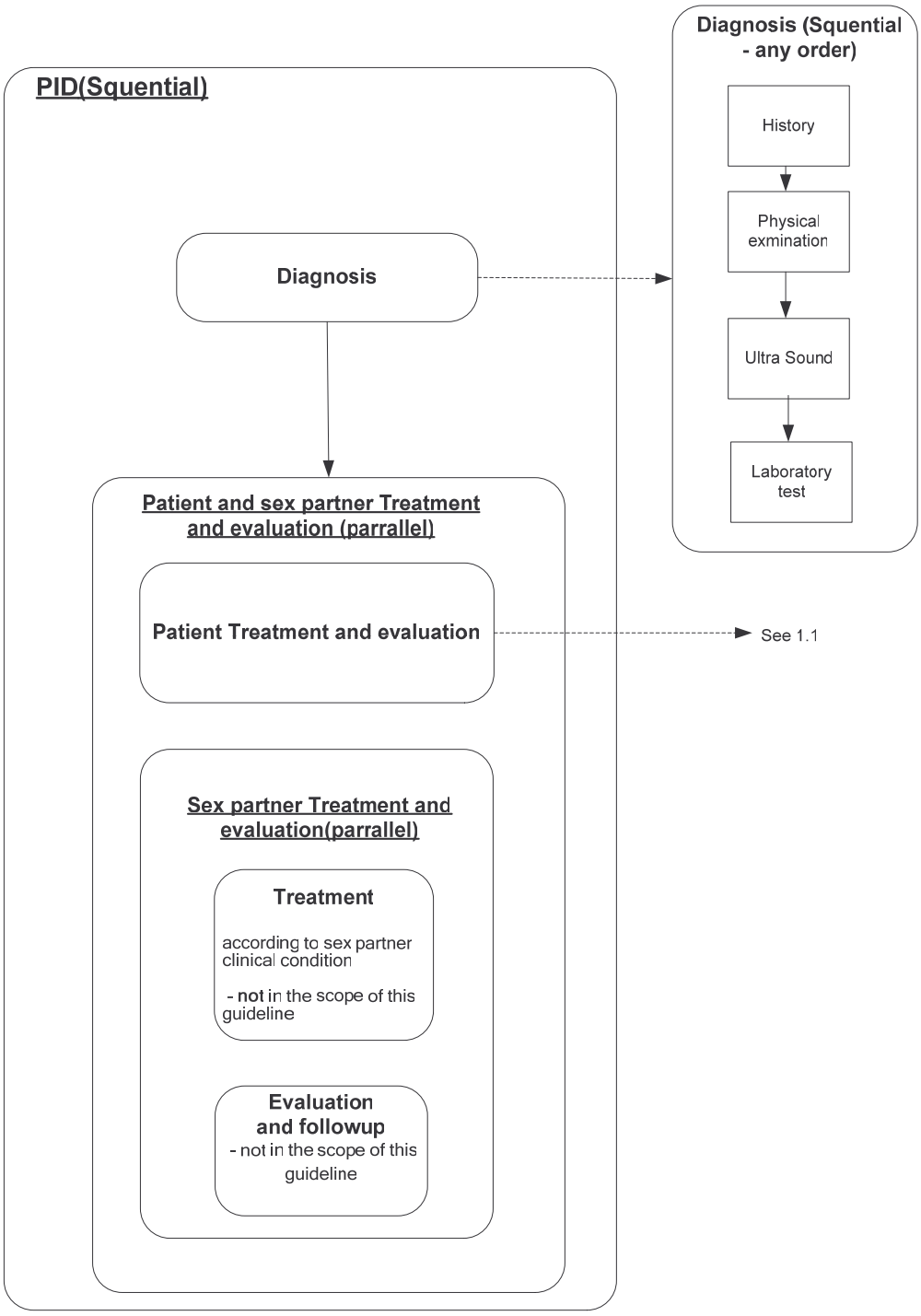
IV ofloxacin has been investigated as a single agent; however because of concerns regarding its spectum, metronidazole may be included in the regimen. Preliminary data suggest that levofloxacin is as effective as ofloxacin and may be substituted; its single daily dosing makes it advantageous from a compliance perspective (*83*). Ampicillin/sulbactam plus doxycycline has good coverage against *C. trachomatis, N. gonorrhoeae*, and anaerobes and is effective for patients who have tubo-ovarian abscess.

Appendix B.3 - PID Ontology Specific Consensus Document

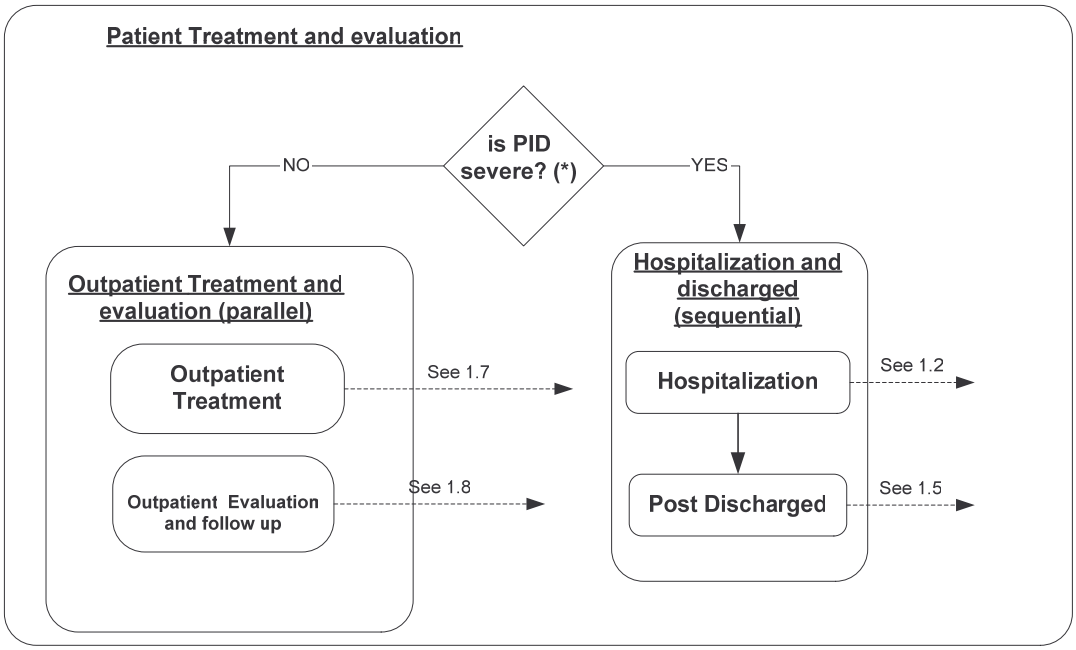
This consensus is based on the following sources (from DeGeL):

- **PID CDC2002** (DeGeL ID - #970) -**Main**
- **CME PID** (DeGeL ID - #972)

3.7. The procedural (clinical) structure of the guideline :

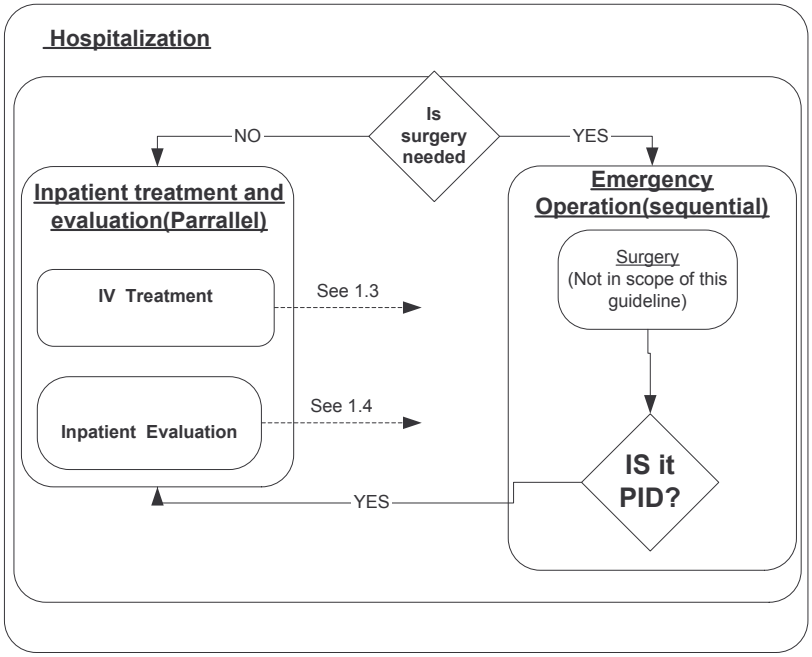


1.1. Patient Treatment and evaluation

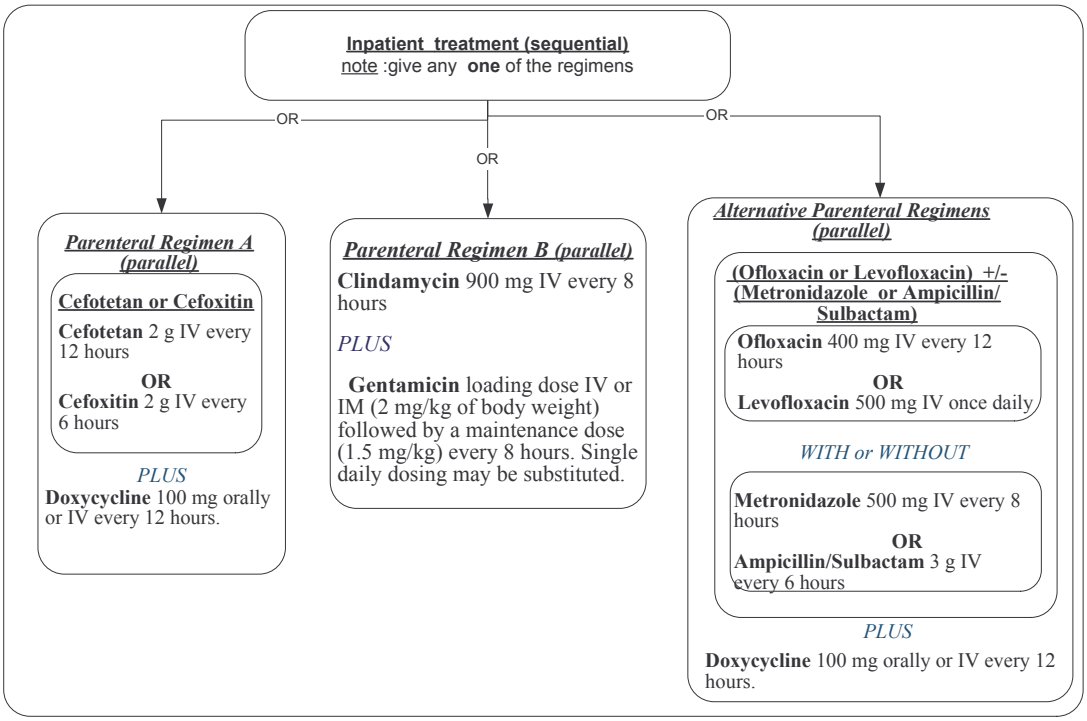


s

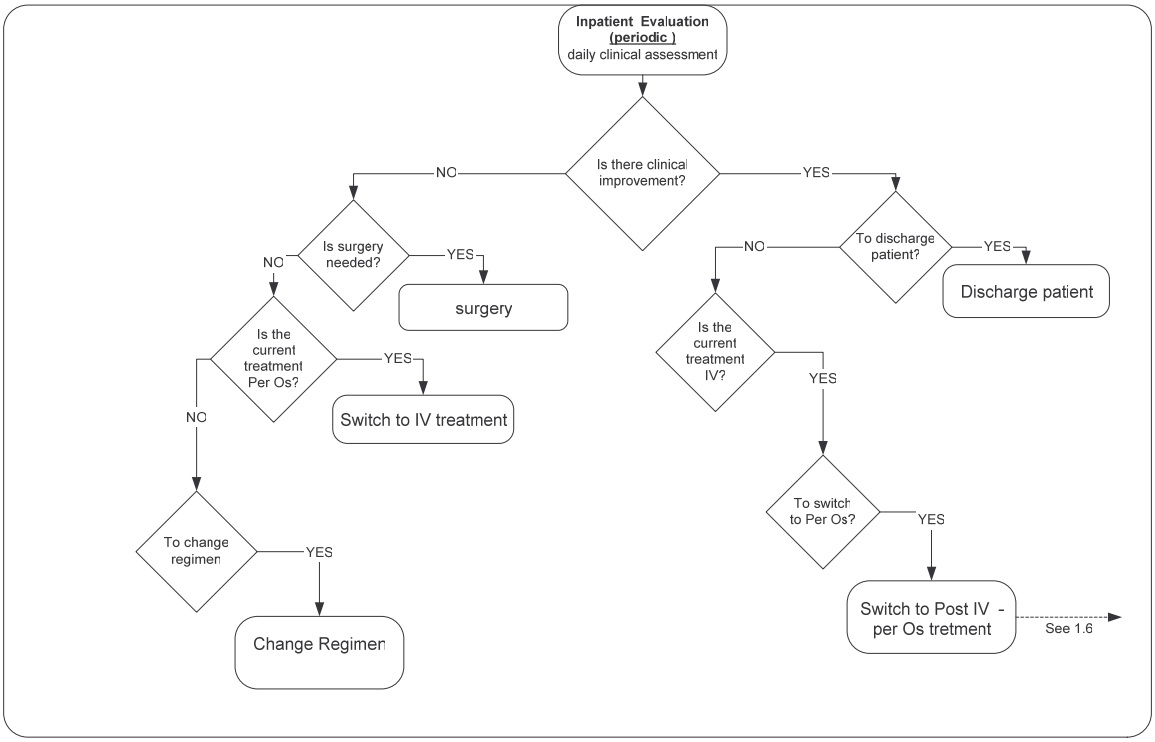
1.2. Hospitalization



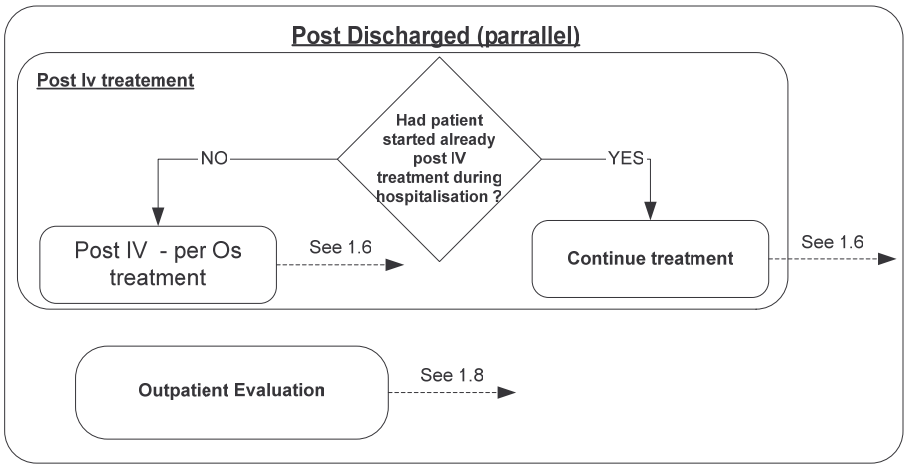
1.3. IV treatment



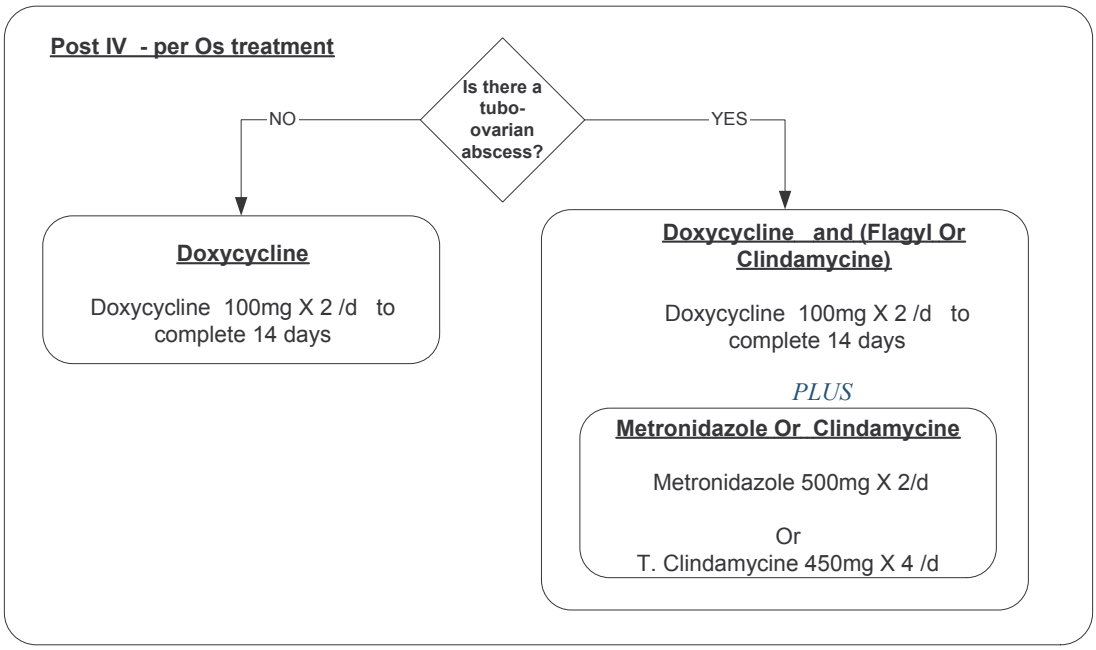
1.4. Inpatient Evaluation



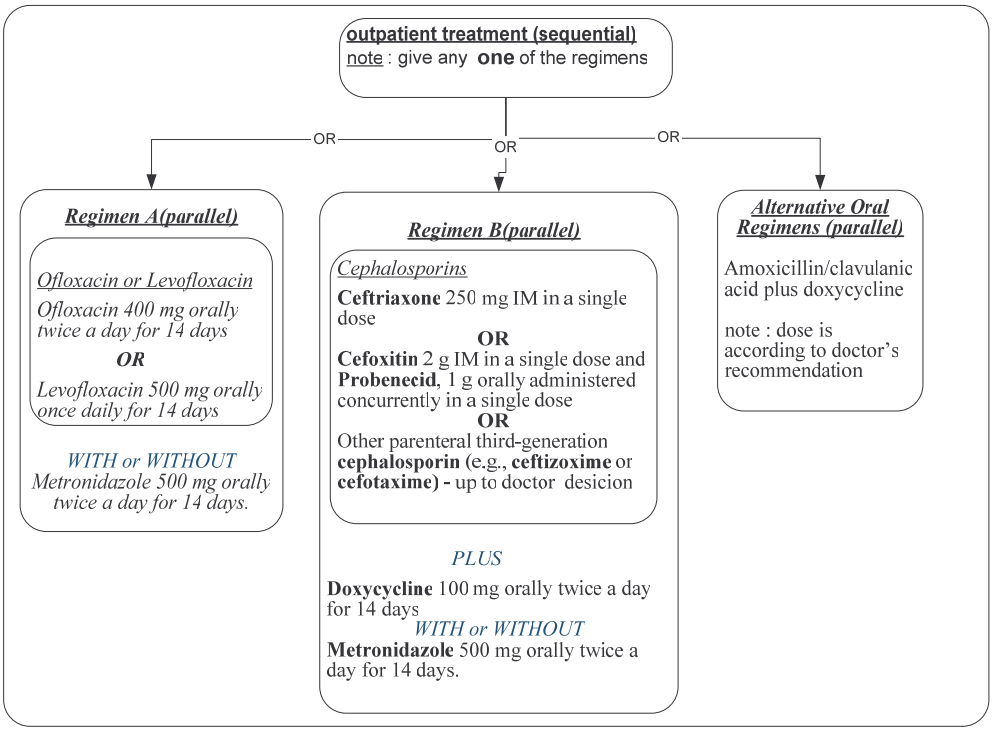
1.5. Post Discharged



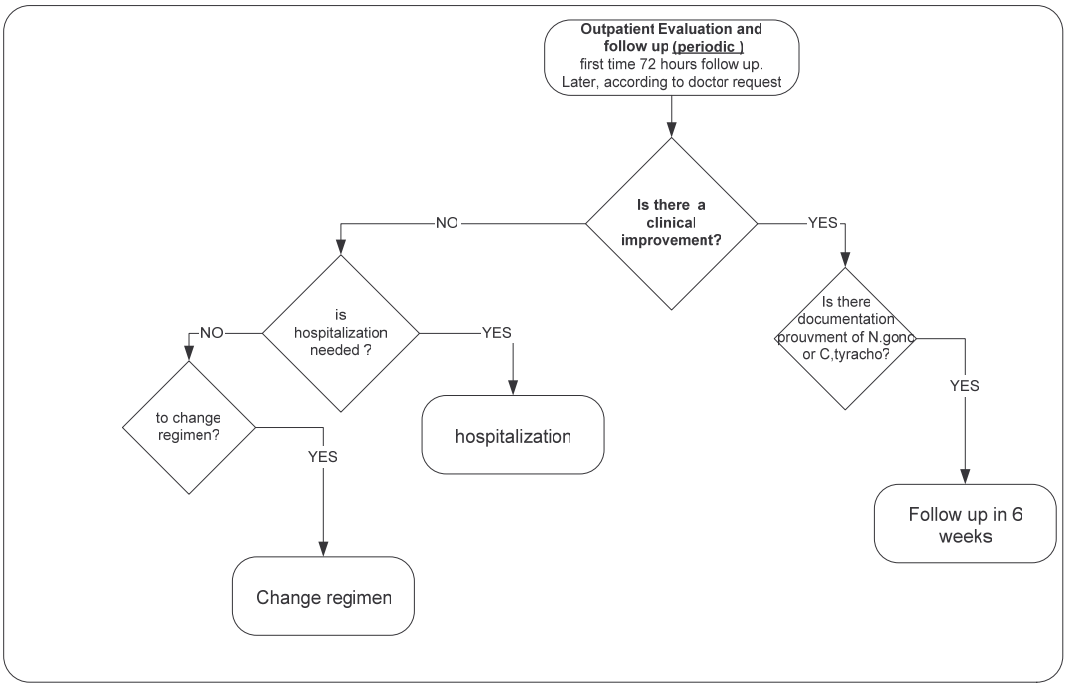
1.6. Post IV - per Os treatment



1.7. Outpatient Treatment



1.8. outpatient Evaluation and follow up



3.8. The declarative part of the steps of the guideline:

Top level - PID	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - Gynecologist clinics
Clinical context	ER, ward , OR,
Intentions *Overall - outcome*	To Cure PID
Filter Condition	(suspected pid –(*)) **and** (sexually active female)
Set Up Condition	?
Abort Condition	Elimination of diagnosis of PID **Or** patient died
Complete Condition	No further treatment (IV or Per Os or surgery) **or** follow up is needed

Diagnosis	
Intentions Overall - process	• To Obtain history • To Obtain physical signs • To Obtain laboratory tests results • To Obtain sonography
Intentions Overall - outcome	• To diagnose patient disease

Hospitalization	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - ward's Gynecologist
Clinical context	ward or operation room
Intention *intermediate process*	To treat severe PID (*)
Filter Condition	Hospitalization criteria(*)
Set Up Condition	
Abort Condition	
Complete Condition	discharge from hospital

Emergency operation	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - Gynecologist ward
Clinical context	operating room
Intention *intermediate process*	To operate patient
Intention *overall outcome*	To reinforce/exelude diagnosis of PID or to excise purulent tissue.
Filter Condition	Surgical emergencies **or** uncertain diagnosis of pid
Set Up Condition	
Abort Condition	
Complete Condition	End of operation

IV treatment	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors – gynecological ward
Clinical context	Ward
Intention *intermediate process*	To administer IV drug according to one of the regimens A,B, Alternative
Intention *overall outcome*	To achieve substantial clinical improvement (*) **OR** To allow PO treatment
Filter Condition	
Set Up Condition	
Abort Condition	
Complete Condition	Substantial clinical improvement (*) **OR** Post IV PO treatment started

inpatient treatment and evaluation	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - ward's Gynecologist
Clinical context	ward
Intention *intermediate process*	To evaluate patient condition and to monitor treatment
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend Condition	Operation needed. **On Suspend** : Emergency operation
reactivate Condition	Operational proved PID.
Complete Condition	

For each Regimen : A,B,Alternative	
Level of evidence	?
Strength of recommendation	?
Actors	
Clinical context	
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition (for each regimen)	• patient develop sensitivity of regimen **OR** • different bacterial sensitivity from empiric treatment **OR** • failure of regimen to help patient
Complete Condition	

Switch to IV treatment	
Level of evidence	?
Strength of recommendation	?
Actors	
Clinical context	
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Switch to Post VI treatment started
Complete Condition	

Switch to Post VI treatment	
Level of evidence	?
Strength of recommendation	?
Actors	
Clinical context	
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Switch to VI treatment started
Complete Condition	

inpatient evaluation	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - Ward's Gynecologist
Clinical context	Ward
Intention *intermediate process*	Evaluation of patient condition and to monitor patient treatment
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Complete Condition	

Discharged	
Level of evidence	?
Strength of recommendation	?
Actors	Referring doctor
Clinical context	Ambulatory clinic
Intention *intermediate process*	to initiate or continue antibiotic treatment to discharged patient
Filter Condition	Patient is discharged
Set Up Condition	
Abort Condition	Failure of treatment or worsening of PID **Or** surgical emergency **ON Abort** : Hospitalization
Complete Condition	

Post IV - per Os treatment	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - ward's Gynecologist
Clinical context	ward
Intention *intermediate process*	to continue the treatment after hospitalization
Intention *overall outcome*	
Filter Condition	substantial clinical improvement (*)
Set Up Condition	
Abort Condition	• patient develop sensitivity of regimen **OR** • different bacterial sensitivity from empiric treatment **OR** • failure of regimen to help patient

Each drug in Post IV - per Os treatment	
Level of evidence	?
Strength of recommendation	?
Actors	
Clinical context	
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition (for each drug)	• not sensitivity to drugs **AND** • not bacteriology sensitivity **AND** • not previous failure **AND** • availability **AND** • patient compliance **AND** • cost of drug
Set Up Condition	
Abort Condition (for each regimen	
Complete Condition	

outpatient treatment and evaluation	
Level of evidence	?
Strength of recommendation	?
Actors	Ambulatory Doctors - Gynecologist
Clinical context	Ambulatory care
Intention *intermediate process*	to initiate or continue antibiotic treatment according to patient condition
Intention *overall outcome*	to resolve mild pid
Filter Condition	Patient has mild pid **Or** patient discharged
Set Up Condition	
Abort Condition	Failure of treatment or worsening of PID **Or** surgical emergency **ON Abort** : if hospitalization needed then hospitalization plan, else if need to change regimen- start **outpatient treatment and evaluation** again
Complete Condition	

Outpatient treatment	
Level of evidence	?
Strength of recommendation	?
Actors	Ambulatory Doctors - Gynecologist
Clinical context	Ambulatory care
Intention *intermediate process*	to treat with one of the regimens
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Complete Condition	End of the treatment time period

outpatient evaluation and follow up	
Level of evidence	?
Strength of recommendation	?
Actors	Ambulatory Doctors - Gynecologist
Clinical context	Ambulatory care
Intention *intermediate process*	Evaluation of patient condition and to monitor patient ambulatory treatment
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Complete Condition	

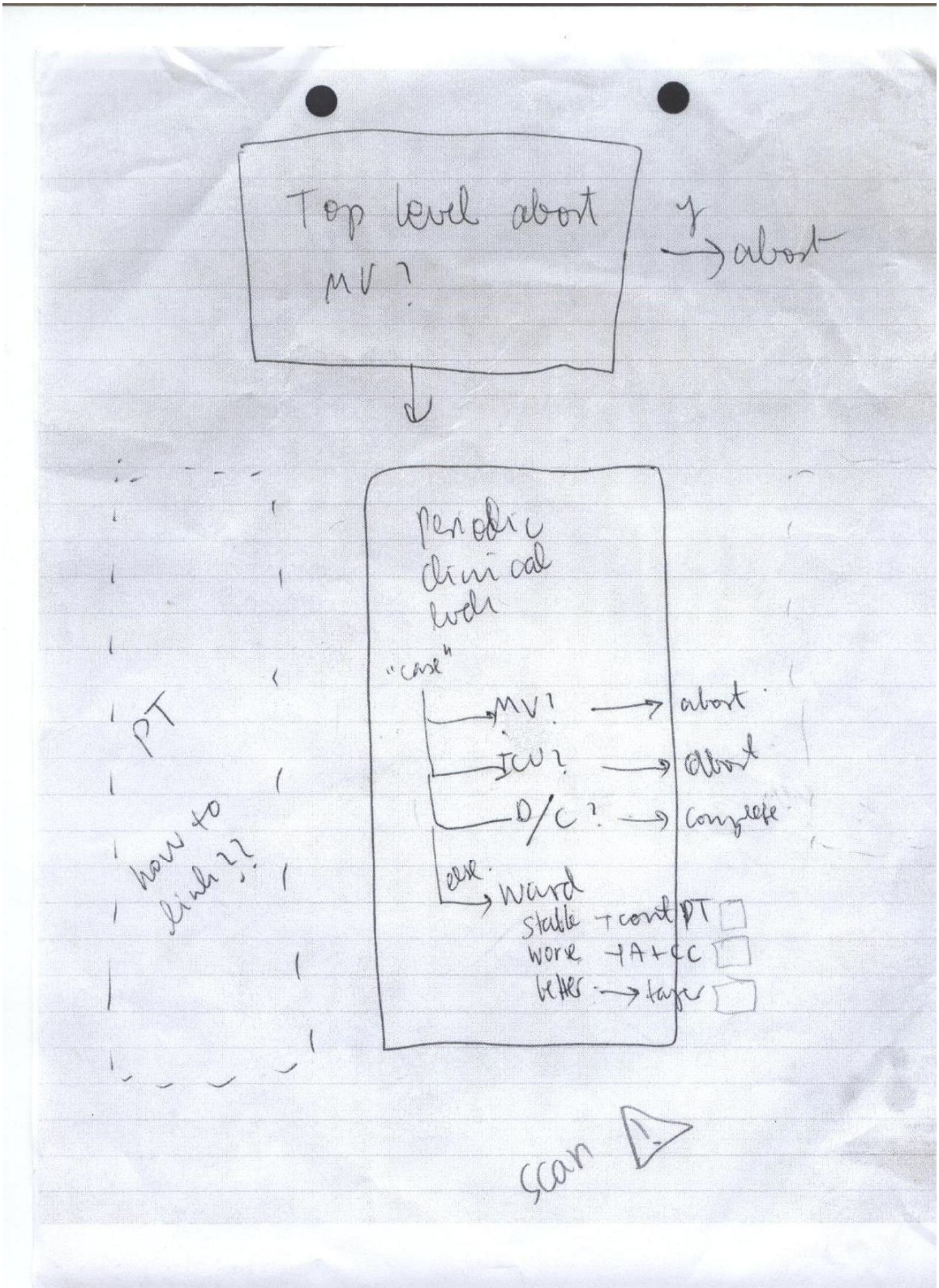
Regimens A,B,Alternative	
Level of evidence	?
Strength of recommendation	?
Actors	
Clinical context	
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition (for each regimen)	•
Set Up Condition	
Abort Condition (for each regimen)	• patient develop sensitivity of regimen **OR** • different bacterial sensitivity from empiric treatment **OR** • failure of regimen to help patient
Complete Condition	

For Each drug : Regimens A,B,Alternative	
Level of evidence	?
Strength of recommendation	?
Actors	
Clinical context	
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition (for each drug)	• not sensitivity to drugs **AND** • not bacteriology sensitivity **AND** • not previous failure **AND** • availability **AND** • patient compliance **AND** • cost of drug
Set Up Condition	
Abort Condition (for each regimen	
Complete Condition	

Sex Partner Treatment and evaluation	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - Urologist , Gynecologist , family physician
Clinical context	Ambulatory / hospital
Intention *intermediate process*	Evaluation of sex partner clinical condition and to monitor patient treatment
Intention *overall outcome*	
Filter Condition	had sexual contact with the patient during the last 60 days
Set Up Condition	
Abort Condition	
Complete Condition	Partner Clinical improvement

Follow Up to patient and sex partner	
Level of evidence	?
Strength of recommendation	?
Actors	Doctors - Urologist , Gynecologist
Clinical context	Ambulatory clinic
Intention *intermediate process*	
Intention *overall outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Recurrence of disease during follow up time period **ON ABORT** : patient and partner treatment and evaluation
Complete Condition	Negative screening after 6 weeks

Appendix B.4 – The first draft of creating the procedural part of the consensus of the COPD GL



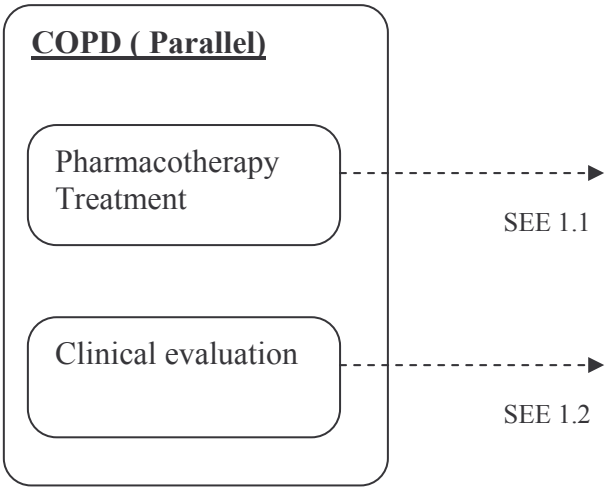
Appendix B.5 – The COPD Ontology Specific Consensus Document

This consensus is based on the following sources (from DeGeL):

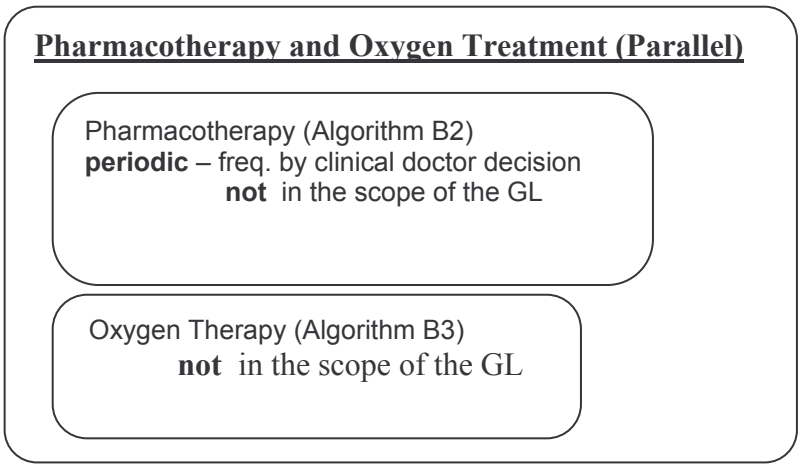
- COPD annotation ED and Ward (DeGeL ID - #929) -**Main**
- Algorithm ED and Ward (DeGeL ID - #935)
- Gold – Standard - #3983

1.1 procedural knowledge

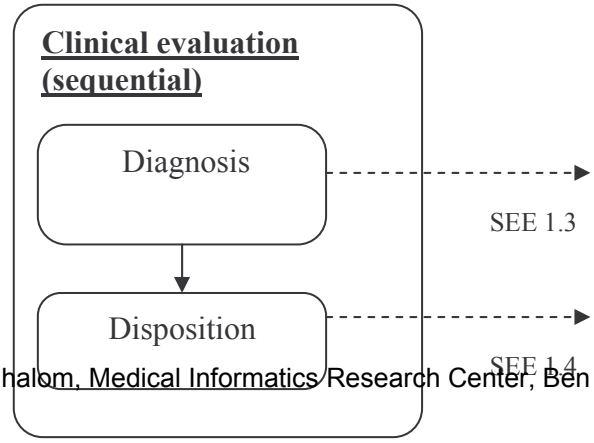
Main Root



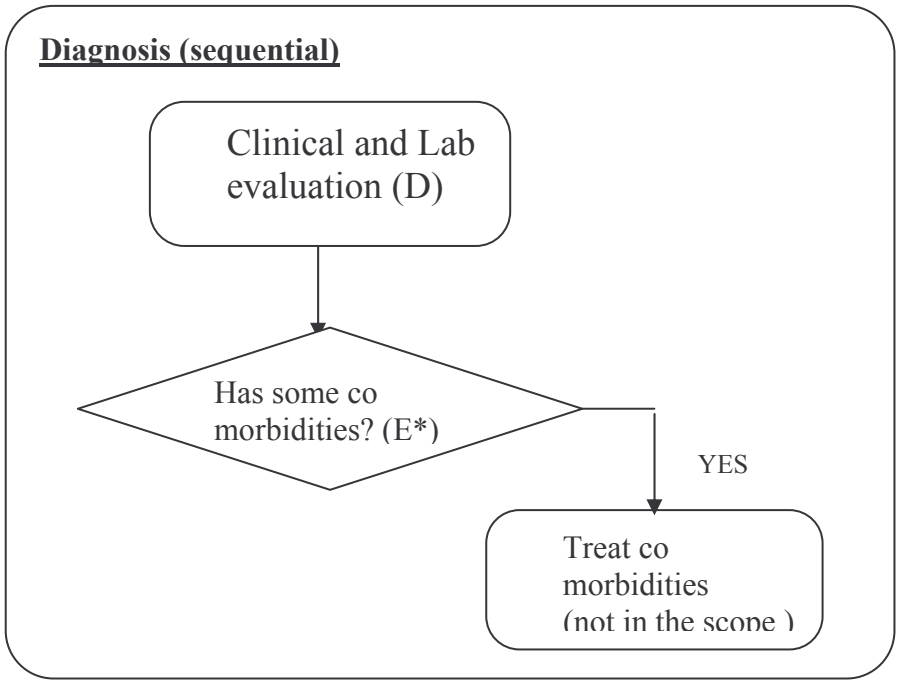
1.1 Pharmacotherapy and Oxygen Treatment



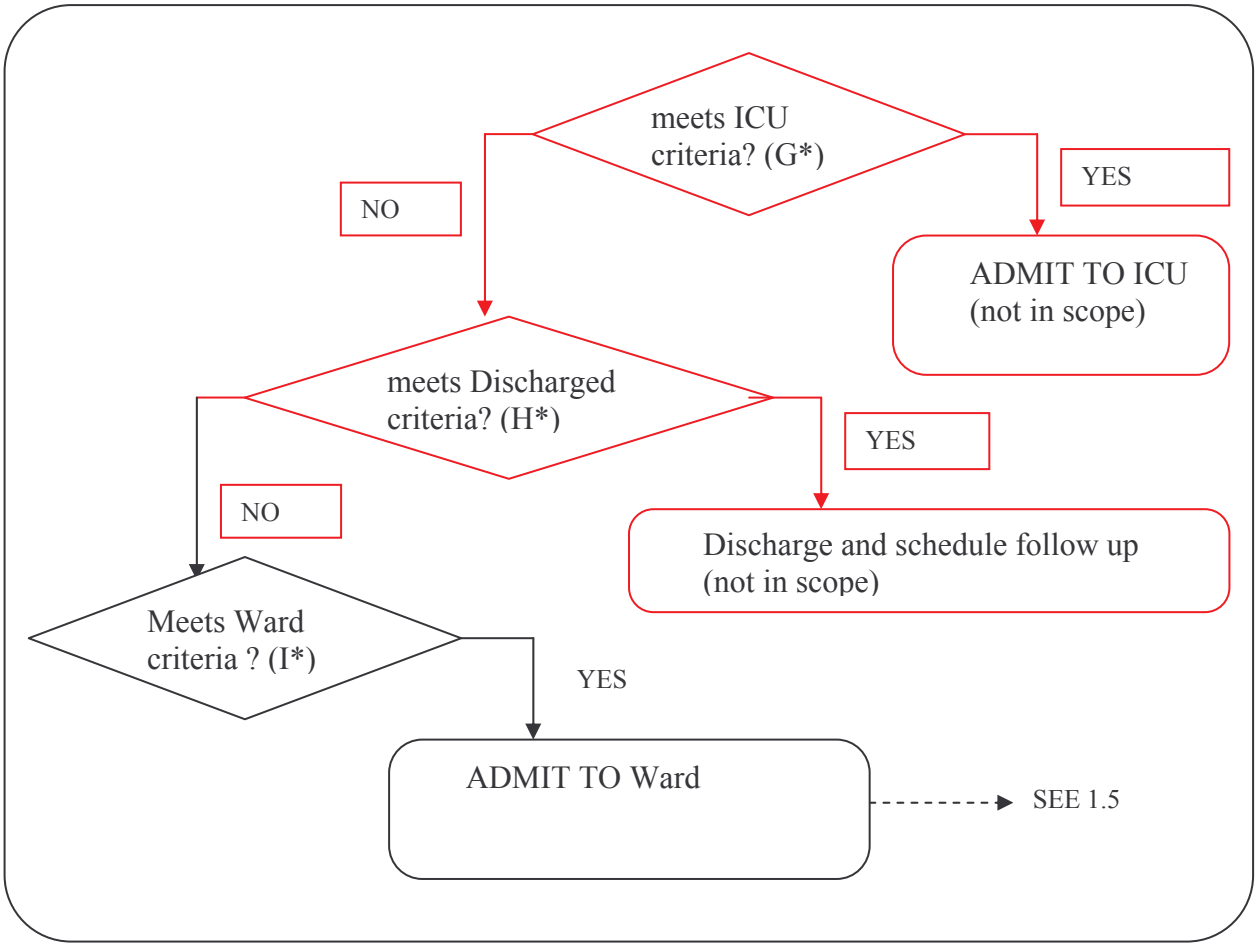
1.2 Clinical evaluation



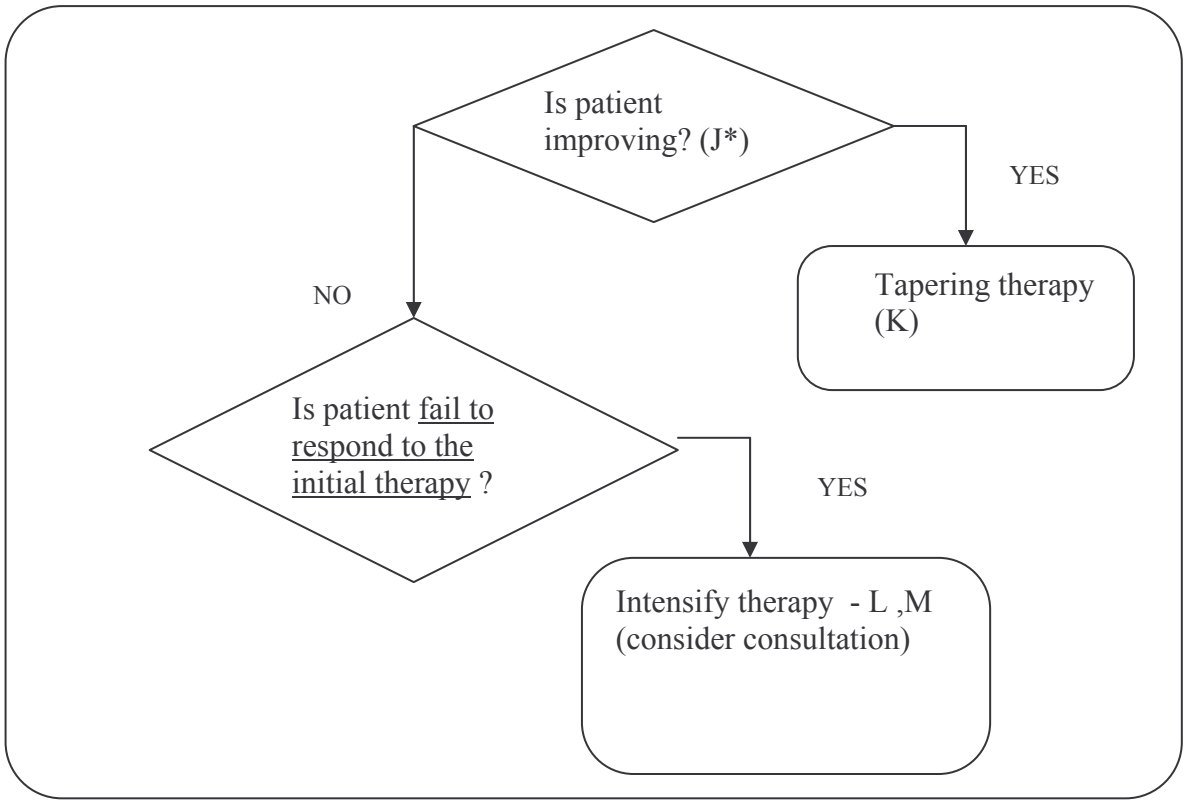
1.3 Diagnosis



1.4 Dispositions



1.5 ADMIT to Ward (periodic) - frequency to doctor's decision



1.2 The declarative part of the steps of the guideline:

Top level - COPD	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	1.Features of the severe exacerbation are resolved (see Annotation D). 2.Anticipated need for inhaled bronchodilators is not more frequent than every 4 hours and the patient is on oral medication. 3.Reversible component of airway obstruction, if present, is under stable control. 4.Patient or caregiver understands appropriate use of medications. 5.Follow-up and home care arrangements have been completed (e.g., visiting nurse, oxygen delivery, meal provisions). 6.Patient, family, and physicians are confident that the patient can manage successfully.
Intentions *intermediate - process*	5. Decrease frequency of inhaled beta$_2$-agonists to every 4 to 6h AND 6. Switch to MDI with spacers AND 7. Switch from parenteral to oral medication AND 8. Titrate oxygen as per oxygen protocol
Intentions *intermediate - outcome*	avoid complications auto PEEP AND avoid acute respiratory alkalosis and Improvement (indicated by Reduced dyspnea AND Decreased respiratory rate AND Improved air movement AND Decreased use of accessory muscles AND Improved peak expiratory flow AND Improved FEV_1 and/or ABGs.
Filter Condition	(underlying COPD) AND (acute exacerbation COPD(A*)-defined in guideline knowledge, presence of any of items)
Set Up Condition	None
Abort Condition	Patient need MV(B*) OR does not meet advance directives (patient's wishes) OR CPR OR ADMIT TO ICU(F) or death
Suspend condition	
Reactivate condition	
Complete Condition	Discharge from ward
Effects	
Guideline knowledge	**sections A,B,D,E,I and G**

Pharmacotherapy and Oxygen Treatment	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Is Patient improving or patient fail to respond to the initial therapy?
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Clinical evaluation	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Is Patient ADMIT TO ICU ?
Suspend condition	
Reactivate condition	
Complete Condition	Is Discharge from ward?
Effects	
Guideline knowledge	**D,E,H,I,j,k,L.M**

Pharmacotherapy	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	**B2**

Oxygen therapy	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	**B3**

Diagnosis	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Clinical and Lab evaluation	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	**D**

Is **patient has co morbidities**	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED and ward
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	**E**

Dispositions	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	Is Patient ADMIT TO ICU ?
Set Up Condition	
Abort Condition	
Suspend condition	Is Discharge from ward?
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

ADMIT to Ward	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	Discharge from ward
Effects	
Guideline knowledge	

Intensify therapy	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Is Patient improving or patient fail to respond to the initial therapy?
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	**L,M**

Tapering therapy	
Level of evidence	N/A
Strength of recommendation	N/A
Actors	point of care physician ED and wards
Clinical context	ED
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Set Up Condition	
Abort Condition	Is Patient improving or patient fail to respond to the initial therapy?
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	**J**

Appendix B.6 – The Hypothyroidism Ontology specific Consensus Document

1. *Clinical Consensus*

This consensus is based on the following sources (from DeGeL):

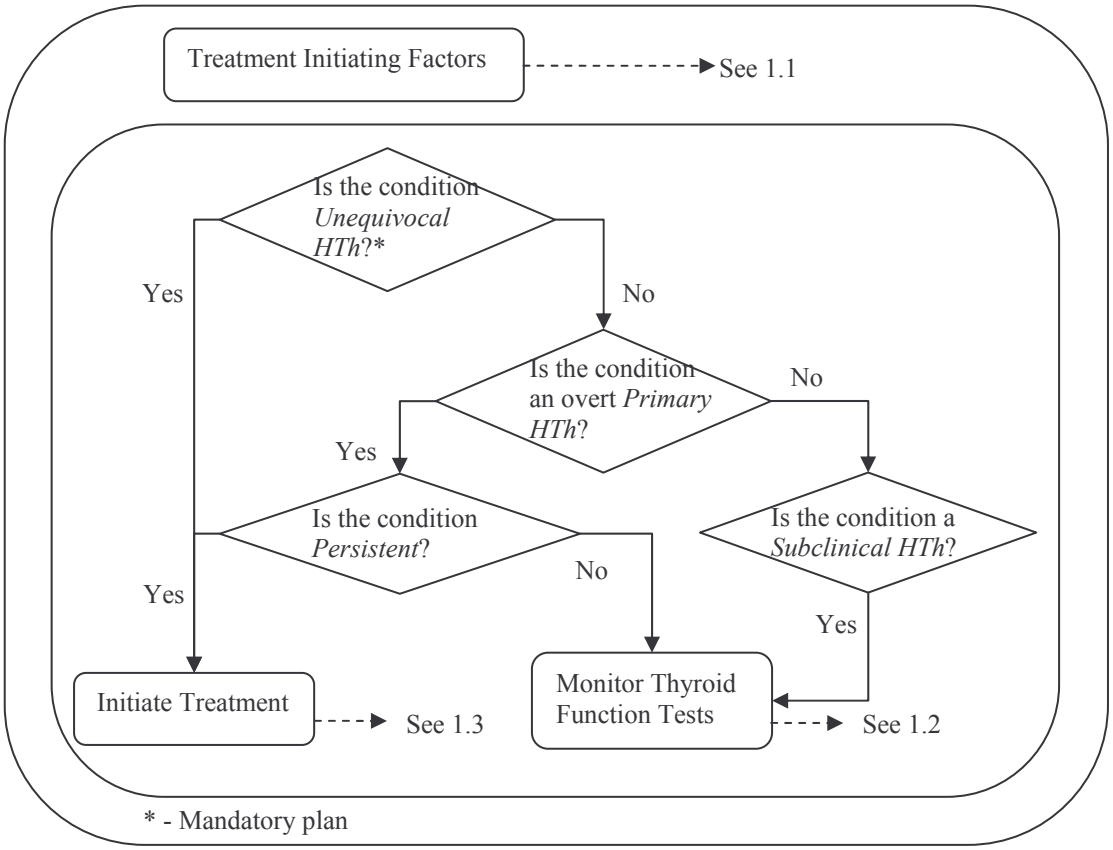
- **AACE Thyroid Task Force 2002** (DeGeL ID - #973)
- The authors of the guideline decided to focus only on the treatment (leaving diagnosis out) of primary (no secondary or other types) hypothyroidism (HTh).

2. *Ontology-Specific Consensus (Hybrid-Asbru)*

Following is a detailed description of the procedural structure and declarative knowledge of the Hypothyroidism (HTh) guideline.

2.1. *The procedural structure of the guideline*

Plan #1: Root plan (parallel)

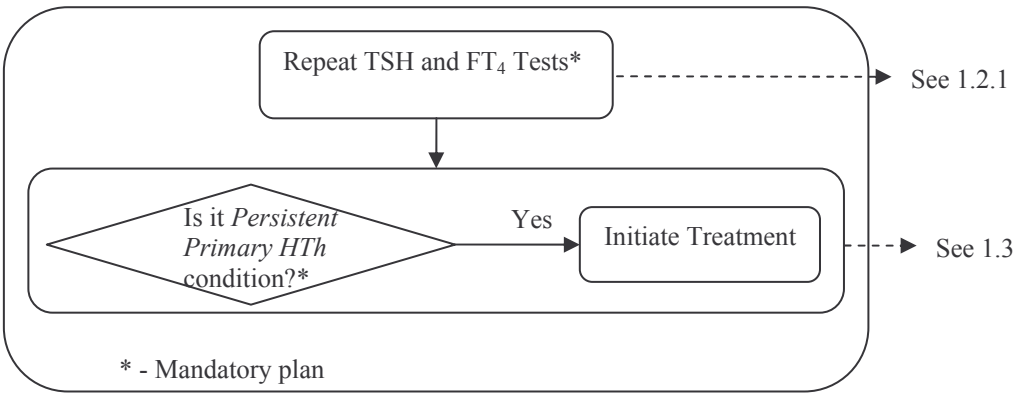


Plan #1.1: Treatment Initiating Factors

Consider the following factors:

- Treatment initiating factors:
- Family history of hyper- of hypothyroidism
- Personal history of thyroid surgery
- Personal history of neck irradiation
- Personal history of infiltrative disease (e.g. amyloidosis, Wilson's, hemochromatosis)
- Positive anti-thyroid antibodies titer
- History of amiodarone treatment (until less than 6 months)
- Enlarged thyroid gland or thyroid nodules (by palpation or ultrasound)
- Personal history of another autoimmune disorder, e.g.: vitiligo, rheumatoid arthritis, SLE, diabetes mellitus (including type 2), pernicious anemia, Addison, Celiac

Plan #1.2: Monitor Thyroid Function Tests (sequential)



Plan #1.2.1: Repeat TSH and T_4 (cyclical)

Check TSH and FT_4 values

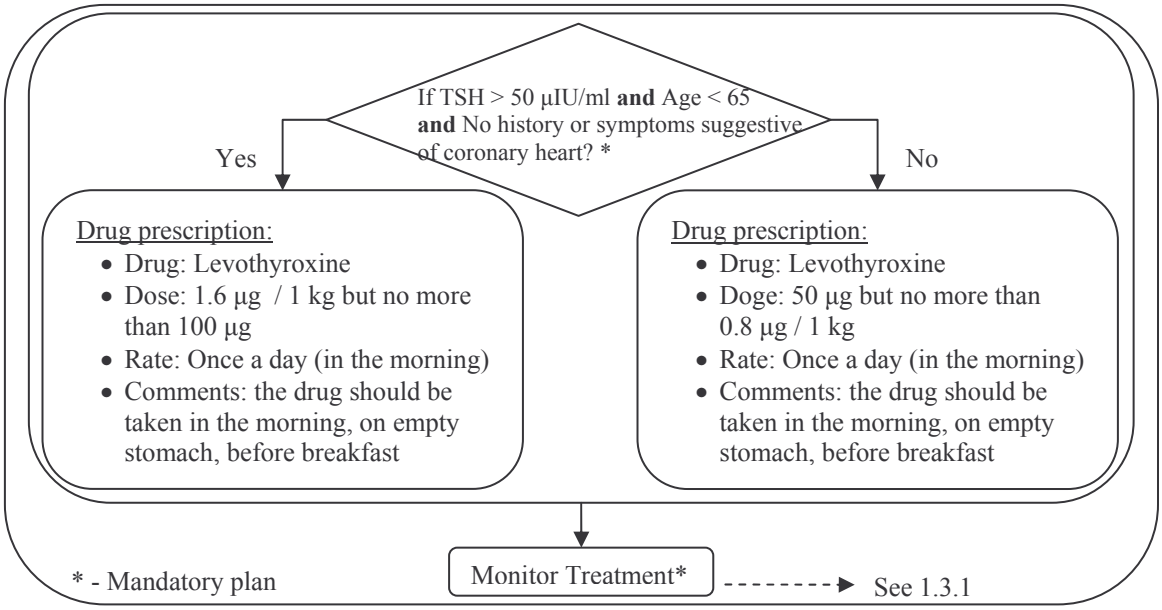
Start specification:
At least 2 month after the last check of TSH and FT_4
Repeat specification:
Depends on the check results:

- *Subclinical* HTh: if three last checks are *subclinical*, in 6 month, else in 2 month (establishing a persistent condition)
- *Normal* function: in six month (establishing a normal function of the Thyroid gland)

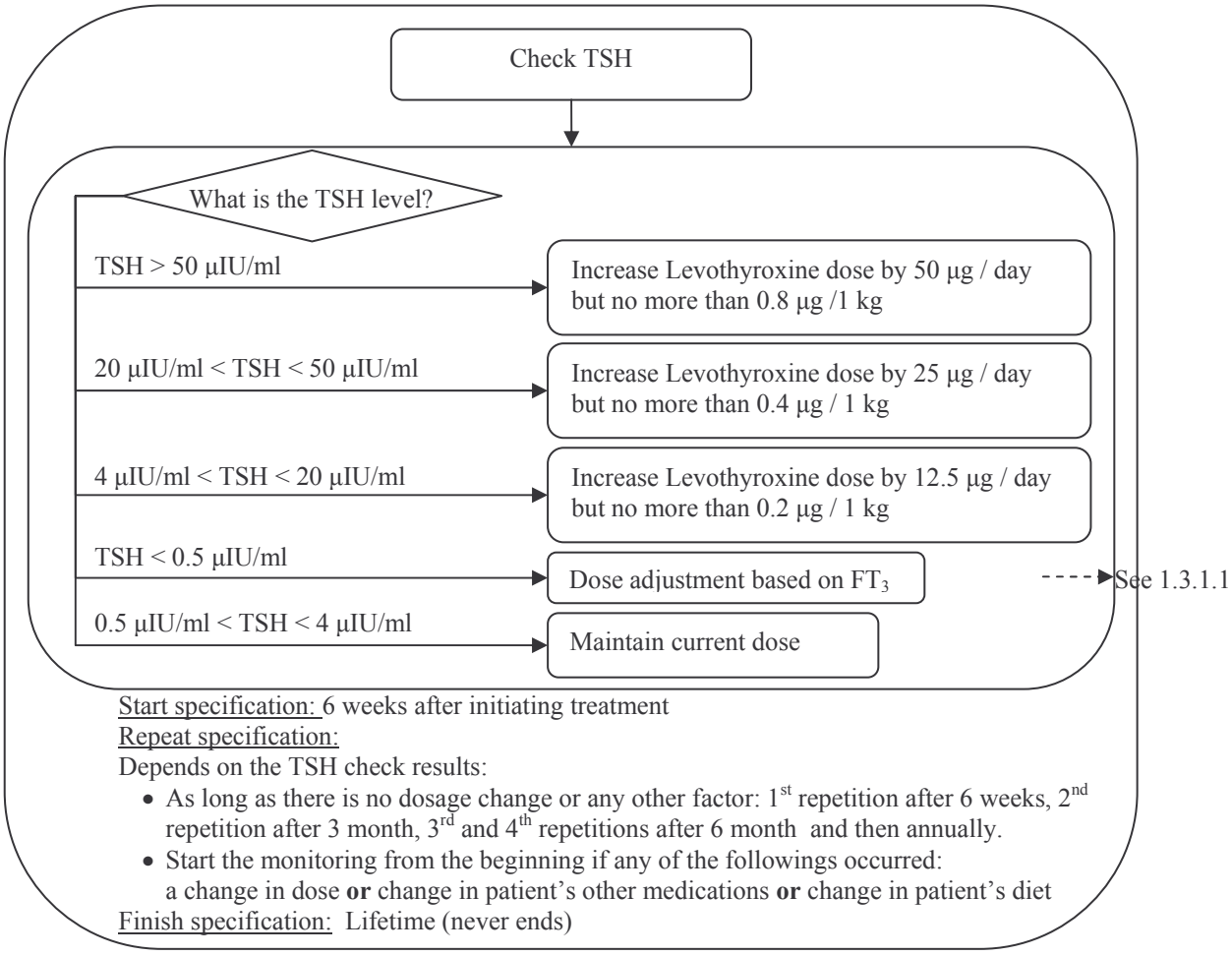
Finish specification:
Depends on the check results:

- Two sequent checks with normal results of TSH and FT_4
- The checks indicate an overt primary HTh *Persistent* condition

Plan #1.3: Initiate Treatment (sequential)



Plan #1.3.1: Monitor Treatment (cyclical)



Start specification: 6 weeks after initiating treatment

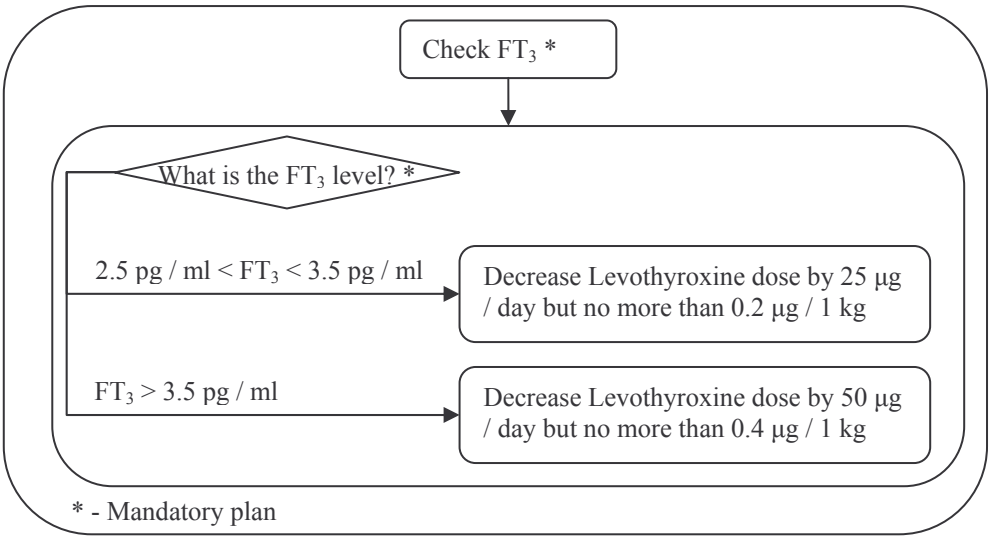
Repeat specification:

Depends on the TSH check results:

- As long as there is no dosage change or any other factor: 1st repetition after 6 weeks, 2nd repetition after 3 month, 3rd and 4th repetitions after 6 month and then annually.
- Start the monitoring from the beginning if any of the followings occurred: a change in dose **or** change in patient's other medications **or** change in patient's diet

Finish specification: Lifetime (never ends)

Plan #1.3.1.1: Dose adjustment based on FT_3 (sequential)



2.2. The declarative knowledge of the guideline

Plan #1: Root Plan – Hypothyroidism (HTh)	
Level of evidence	
Strength of recommendation	
Actors	Primary physicians
Clinical context	Outpatient clinic
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	Maintain monitoring of thyroid function tests
Intentions *intermediate – outcome*	Achieve Euthyroid (the state of having normal thyroid hormonal activity) patients
Filter Condition	Age > 18 (adult) **and** Not pregnant **and** (patient complains seem compatible with HTh **or** Has a TSH test performed in the last 2 month with a value > 4.1 μIU/ml) **and** (no Hypothalamic disease **or** no Hypophisial disease) **and** no thyroid hormone resistance syndrome
Setup Condition	TSH and T_4 lab tests results taken no longer than 2 month before.
Abort Condition	Death **or** Patient diagnosed with thyroid cancer **or** Pregnancy **or** non-compliant patient **or** non-thyroid disease which may affect thyroid function
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	• Concept 1: TSH: normal - 0.5 μIU/ml ≤ TSH ≤ 4 μIU/ml; hypo – TSH ≥ 4.1 μIU/ml • Concept 2: FT_4 : normal – 0.8 ng/dl < FT_4 < 1.5 ng/dl • Concept 3: FT_3 : normal – 2.3 pg/ml < FT_3 < 4.2 pg/ml • Concept 4: Unequivocal HTh - Characteristic clinical symptoms and or signs for HTh of more than one month duration with TSH > 10 μIU/ml and T4 < 0.8 μg/ml and at least two of the symptoms described in the guideline source under clinical features (pp.463) • Concept 5: Hyperthyroidism - (TSH ≤ 0.4 μIU/ml and FT_3 > 4.2 pg/ml) • Concept 6: Primary HTh – (TSH > 10 μIU/ml) or (TSH > 4.1 μIU/ml and T_4 < 0.8 μg/ml) or (TSH ≥ 4.1 μIU/ml and T_4 > 0.8 μg/ml and *Imitating factors*) • Concept 7: Subclinical HTh - (4.1 μIU/ml ≤ TSH < 10 μIU/ml and T_4 > 0.8 μg/ml) • Concept 8: Persistent HTh Condition – At least two TSH and T_4 tests taken two months apart, which reveals a persistent increase of TSH or fix high TSH and a persistent decrease of T_4 or fix low T_4. • Concept 9: Typical complains of patients with HTh (p. 463, clinical features).

Plan #1.1: Treatment Initiating Factors	
Level of evidence	
Strength of recommendation	
Actors	
Clinical context	
Intentions *Overall - process*	
Intentions *Overall - outcome*	Increase the probability that the patient has persistent Hypothyroidism
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Setup Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Plan #1.2: Monitor Thyroid Function Tests	
Level of evidence	
Strength of recommendation	
Actors	
Clinical context	
Intentions *Overall - process*	
Intentions *Overall - outcome*	To establish the need for thyroid replacement therapy
Intentions *intermediate - process*	Miantain monitoring of TSH and FT4 tests
Intentions *intermediate - outcome*	
Filter Condition	
Setup Condition	

Abort Condition	It's not a persistent Hypothyroidism
On Abort	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Plan #1.2.1: Repeat TSH and T_4	
Level of evidence	
Strength of recommendation	
Actors	
Clinical context	
Intentions *Overall - process*	To acieve well-based disgnosis of persistent Hypothroidism
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Setup Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Plan #1.3: Initiate Treatment	
Level of evidence	
Strength of recommendation	
Actors	
Clinical context	
Intentions *Overall - process*	Provide initial dose
Intentions *Overall - outcome*	To provide Levothyroxine replacement dose that will normalize TSH in a hypothyroid patient
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Setup Condition	
Abort Condition	TSH < 0.5 μIU/ml **and** $FT_3 < 2.5$ pg / ml
Suspend condition	
On-Abort	Consider referral to an endocrinologist
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Plan #1.3.1: Monitor Treatment	
Level of evidence	
Strength of recommendation	
Actors	
Clinical context	
Intentions *Overall - process*	
Intentions *Overall - outcome*	
Intentions *intermediate - process*	Maintain monitoring of levothyroxine replacment dose
Intentions *intermediate - outcome*	Achieve and maintain normal TSH
Filter Condition	

Setup Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Plan #1.3.1.1: Dose adjustment based on FT_3	
Level of evidence	
Strength of recommendation	
Actors	
Clinical context	
Intentions *Overall - process*	Adjust Levothyroxine dose based on FT_3
Intentions *Overall - outcome*	
Intentions *intermediate - process*	
Intentions *intermediate - outcome*	
Filter Condition	
Setup Condition	
Abort Condition	
Suspend condition	
Reactivate condition	
Complete Condition	
Effects	
Guideline knowledge	

Appendix C – Questioners

Appendix C.1 - Questioner 1 -

- Purpose - to check the user's attitude regarding making an ontological consensus i.e. a consensus which composed from a clinical consensus (e,g, diagnosis , treatment) and a specific guideline specification language semantics (e.g. filter condition of the treatment
- Target participants - Medical experts (MEs) , knowledge engineers(KE) which participated in the process of making the ontological consensus ach of them will fill a questioner

Instructions - In what way, every of the following subjects helped you to **make an ontological consensus**

- Grades : -3 interrupted a lot
  0 didn't contribute at all
  3 very much contributing

subject	Description	Example	-3	-2	-1	0	1	2	3
Knowing the multiple representation	Acquired The knowledge of the guideline in 3 steps	1.Structured text 2. semi-formal 3.formal							
Ontology	key concepts, properties and their relations	Asbru ontology							
Asbru Krs –Procedural part	The clinical pathway described as flow chart (The first part of the ontological consensus)	The guideline composed from 3 parallel steps – diagnosis, treatment and follow up)							
Asbru Krs –Declerative part	Each plan has some attributes describing the Asbru semantics of it	The filter or abort condition of the treatment plan. Each plan has table in the ontological consensus							
DeGeL	The digital guideline library	The library of the guidelines we developed							
URUZ –main interface	The tool for guideline markup	The main interface to perform markup							
Plan-body wizard	The tool for guideline structuring into tree of plans	Structure the guideline using wizard for different type of plans							
IndexiGuide	The tool for indexing the guideline	index according axis such as : diagnostic findings, specialties ect..							
Vaidurya	The tool for searching the guideline	Search PID guideline with "PID" word in the title							
Vocabulary server	The tool for finding terms In standard vocabularies such as Loinc, CPT	Finding the term "doxycyline" in NDF vocabulary and get its ID							
Spock	The tool for Apply the guidelines	Appling the PID using SPOCK							
your Medical expertise	All your expertise knowledge regarding the guideline	Give a particular regimen , or take some tests							
Reading the guideline sources before making ont. consensus	Reading the textual content of the guideline	The CDC source for PID							
Having more than one source	The markup has more then one source	The PID markup is based the PID + the British source							

Appendix C.2 - Questioner 2 -

- Purpose - to check the GEs attitude regarding understanding Asbru KRs
- Target participants - Guidelines editors (GEs)
- Instructions - How difficult it was for you to **understand** each one of the following KRs before performing markup
- Grading : -3 difficult to understand
  0 not relevant
  3 very easy to understand

subject	Description	Example	-3	-2	-1	0	1	2	3
Knowing the multiple representation	Acquired The knowledge of the guideline in 3 steps	1.Structured text 2. semi-formal 3.formal							
Ontology	key concepts, properties and their relations	Asbru ontology							
Asbru Krs –Procedural part	The clinical pathway described as flow chart (The first part of the ontological consensus)	The guideline composed from 3 parallel steps – diagnosis, treatment and follow up)							
Asbru Krs –Declerative part	Each plan has some attributes describing the Asbru semantics of it	The filter or abort condition of the treatment plan. Each plan has table in the ontological consensus							
DeGeL	The digital guideline library	The library of the guidelines we developed							
URUZ –main interface	The tool for guideline markup	The main interface to perform markup							
Plan-body wizard	The tool for guideline structuring into tree of plans	Structure the guideline using wizard for different type of plans							
IndexiGuide	The tool for indexing the guideline	index according axis such as : diagnostic findings, specialties ect..							
Vaidurya	The tool for searching the guideline	Search PID guideline with "PID" word in the title							
Vocabulary server	The tool for finding terms In standard vocabularies such as Loinc, CPT	Finding the term "doxycyline" in NDF vocabulary and get its ID							
Spock	The tool for Apply the guidelines	Appling the PID using SPOCK							
your Medical expertise	All your expertise knowledge regarding the guideline	Give a particular regimen , or take some tests							
Reading the guideline sources before making ont. consensus	Reading the textual content of the guideline	The CDC source for PID							
Having more than one source	The markup has more then one source	The PID markup is based the PID + the British source							

4

Appendix C.3 Questioner 3-

- Purpose - to check the guidelines editors attitude regarding structuring Asbru KRs
- Target participants - Guidelines editors (GEs)
- Instructions - How difficult it was for you to **structure** each one of the following KRs while performing markup?
- Grades : -3 difficult to structure
  0 not relevant
  3 very easy to structure

KR	Description	Example	-3	-2	-1	0	1	2	3
Level of evidence	the grades of evidenced based	1,1++,2,3,							
Strength of recommendation	the level we of recommendation of this plan	A,B,C							
Actors	Specifies who is responsible or taking part in performing the guideline actions	Nurse, gynecologist, etc.							
Clinical context	Specifies where in the clinical setting the patient is being seen	outpatient clinic, ER, ICU, extended care, etc.							
Intentions *Overall - process*	The **action(s)** that should take place **after finishing the plan**	patient had visited dietitian							
Intentions *Overall - outcome*	achieved, or maintained, or avoided **after finishing the plan**	patient had less than one high							
Intentions *intermediate - process*	place during the **process of the plan**	monitor blood glucose once a							
Intentions *intermediate - outcome*	The **state(s)** that should be achieved, or maintained, or avoided **during the process** of the plan	• avoid complications auto PEEP • avoid acute respiratory alkalosis and							
Filter Condition	Specifies the exclusion/inclusion criteria of the guideline	((Patient must be female) AND (underlying COPD) AND ((acute exacerbation COPD-defined in guideline knowledge))							
Setup Condition	Specifies the additional criteria which should be achieved through actions by the physician prior to the start of plan applying	Patient need to have a positive glucose-tolerance test **AND** hgbA_1C > 4.5							

KR	Description	Example	-3	-2	-1	0	1	2	3
Abort Condition	Specifies when a plan must end unsuccessfully	patient death							
Suspend condition	Specifies when a plan must be put on a hold	Patient's blood glucose has been normal for at least one day							
Reactivate condition	Specifies when a plan can be reactivated after being suspended	Patient's blood glucose has been normal for at least one day							
Complete Condition	Specifies when a plan can end successfully	patient has no symptoms of COPD							
Guideline knowledge	Contains definitions of medical concepts which might be used in several plans such as : Clinical parameters , their definitions , and their classification criteria.	concept names such as: • **sever PID** • **Hospitalization criteria** • **Acute exacerbation of COPD**							
Simple Action	An atomic plan with simple Suitable for defining plans with one • Measure patient temperature	• Give prescription • Check lab test value • Measure patient temperature							
Plan activation	Used to refer a plan which was defined in DeGeL	Every plan which was defined in DeGeL and can be reused							
If-then -else	A condition between 2 plans. When this condition holds ("yes") , then do plan A. when this condition is false do plan B	Is mechanical ventilation required? If yes, then start mechanical ventilation, else start Oxygen therapy							
Switch- case	The criteria has some possible values. For each value, a plan should be defined	if hypertension state is normal do follow-up, if it is mild do treatment if it severe hospitalize patient.							
Repeating plan	A plan that should be repeat more then on time in periods	• BP measure once a day between 1 to 3 weeks							
Subplans – parallel order	plan which is composed from more then one sub plans	There are two or more sub plans which possibly overlap							
Subplans – sequential order	plan which is composed from more then one sub plans	At each moment in time only one plan is performed -							
Subplans – any order	plan which is composed from more then one sub plans	There are two or more sub plans which possibly overlap with no order							
Subplans – unorder	plan which is composed from more then one sub plans	At each moment in time only one plan is performed -with no order							
To be defined	This plan is not in the scope of this guideline, and needed to be define	Emergency operation plan will be defined in the future							

Appendix C.4 Questioner 4-

- Purpose - to check the guidelines editors attitude regarding the different tools they use during the markup process
- Target participants - Guidelines editors (GEs)
- Instructions - In what way, every of the following subjects helped you **when performing markup?**
- Grades : -3 interrupted a lot
  1 didn't contribute at all
  3 very much contributing

KR	Description	Example	-3	-2	-1	0	1	2	3
Level of evidence	the grades of evidenced based	1,1++,2,3,							
Strength of recommendation	the level we of recommendation of this plan	A,B,C							
Actors	Specifies who is responsible or taking part in performing the guideline actions	Nurse, gynecologist, etc.							
Clinical context	Specifies where in the clinical setting the patient is being seen	outpatient clinic, ER, ICU, extended care, etc.							
Intentions *Overall - process*	The **action(s)** that should take place **after finishing the plan**	patient had visited dietitian							
Intentions *Overall - outcome*	achieved, or maintained, or avoided **after finishing the plan**	patient had less than one high							
Intentions *intermediate - process*	place during the **process of the plan**	monitor blood glucose once a							
Intentions *intermediate - outcome*	The **state(s)** that should be achieved, or maintained, or avoided **during the process** of the plan	• avoid complications auto PEEP • avoid acute respiratory alkalosis and							
Filter Condition	Specifies the exclusion/inclusion criteria of the guideline	((Patient must be female) **AND** (underlying COPD) **AND** ((acute exacerbation COPD-defined in guideline knowledge))							
Setup Condition	Specifies the additional criteria which should be achieved through actions by the physician prior to the start of plan applying	Patient need to have a positive glucose-tolerance test **AND** hgbA_1C > 4.5							

KR	Description	Example	-3	-2	-1	0	1	2	3
Abort Condition	Specifies when a plan must end unsuccessfully	patient death							
Suspend condition	Specifies when a plan must be put on a hold	Patient's blood glucose has been normal for at least one day							
Reactivate condition	Specifies when a plan can be reactivated after being suspended	Patient's blood glucose has been normal for at least one day							
Complete Condition	Specifies when a plan can end successfully	patient has no symptoms of COPD							
Guideline knowledge	Contains definitions of medical concepts which might be used in several plans such as : Clinical parameters , their definitions , and their classification criteria.	concept names such as: • **sever PID** • **Hospitalization criteria** • **Acute exacerbation of COPD**							
Simple Action	An atomic plan with simple Suitable for defining plans with one • Measure patient temperature	• Give prescription • Check lab test value • Measure patient temperature							
Plan activation	Used to refer a plan which was defined in DeGeL	Every plan which was defined in DeGeL and can be reused							
If-then -else	A condition between 2 plans. When this condition holds ("yes") , then do plan A. when this condition is false do plan B	Is mechanical ventilation required? If yes, then start mechanical ventilation, else start Oxygen therapy							
Switch- case	The criteria has some possible values. For each value, a plan should be defined	if hypertension state is normal do follow-up, if it is mild do treatment if it severe hospitalize patient.							
Repeating plan	A plan that should be repeat more then on time in periods	• BP measure once a day between 1 to 3 weeks							
Subplans – parallel order	plan which is composed from more then one sub plans	There are two or more sub plans which possibly overlap							
Subplans – sequential order	plan which is composed from more then one sub plans	At each moment in time only one plan is performed -							
Subplans – any order	plan which is composed from more then one sub plans	There are two or more sub plans which possibly overlap with no order							
Subplans – unorder	plan which is composed from more then one sub plans	At each moment in time only one plan is performed -with no order							
To be defined	This plan is not in the scope of this guideline, and needed to be define	Emergency operation plan will be defined in the future							

Appendix C.5 Questioner 5-

- Purpose - to check URUZ usability
- Target participants - Guidelines editors (GEs)
- Instructions – for each question fill the most appropriate grade

	Strongly disagree				Strongly agree
1. I think that I would like to use this system frequently	1	2	3	4	5
2. I found the system unnecessarily complex	1	2	3	4	5
3. I thought the system was easy to use	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system	1	2	3	4	5
5. I found the various functions in this system were well integrated	1	2	3	4	5
6. I thought there was too much inconsistency in this system	1	2	3	4	5
7. I would imagine that most people would learn to use this system very quickly	1	2	3	4	5
8. I found the system very cumbersome to use	1	2	3	4	5
9. I felt very confident using the system	1	2	3	4	5
10. I needed to learn a lot of things before I could get going with this system	1	2	3	4	5

For items 1,3,5,7,and 9 the score contribution is the scale position minus 1. For items 2,4,6,8 and 10, the contribution is 5 minus the scale position. Multiply the sum of the scores by 2.5 to obtain the overall value of SUS

Appendix D – Evaluation Results

Appendix D.1 - Completeness Results

Table 56. The completeness level in each of the Existence groups among the EPs of the PID GL

	KRS	Editor	KRs (In presentage) Exist in GS & not in Markup	KRs (In presentage) Exist in GS and in Markup	KRs (In presentage) Not Exist in GS& Exist in Markup	Total
Context	Actors	EP1		11 (100%)		11.00
		EP2	2 (18%)	9 (82%)		11.00
	Clinical Context	EP1		11 (100%)		11.00
		EP2	2 (18%)	9 (82%)		11.00
	Mean Context	EP1		11 (100%)		
		EP2	2 (18%)	9 (82%)		
		Mean	2 (18%)	10 (91%)		
Intentions	Intermediate Process Intentions	EP1		10 (100%)		10.00
		EP2	2 (20%)	8 (80%)	1 (10%)	10.00
	Overall Outcome Intentions	EP1		4 (100%)		4.00
		EP2		4 (100%)		4.00
	Mean Intentions	EP1		7 (100%)		
		EP2	2 (20%)	6 (90%)	1 (10%)	
		Mean	2 (20%)	6.5 (95%)	1 (10%)	
Conditions	Filter Condition	EP1	1 (4%)	27 (96%)	1 (4%)	28.00
		EP2		26 (93%)	11 (39%)	28.00
	Abort Condition	EP1		10 (100%)		10.00
		EP2		10 (100%)		10.00
	Complete Condition	EP1		5 (100%)		5.00
		EP2	1 (20%)	4 (80%)	1 (20%)	5.00
	Suspend Condition	EP1		1 (100%)		1.00
		EP2		1 (100%)		1.00
	Restart Condition	EP1		1 (100%)		1.00
		EP2		1 (100%)		1.00
	Mean Conditions	EP1	1 (4%)	1 (99%)		
		EP2	1.5 (14%)	8.6 (95%)	4 (30%)	
		Mean	1.3 (9%)	4.8 (97%)	4 (30%)	
PlanBody	PlanBody	EP1		106 (100%)	1 (1%)	106.00
		EP2	3 (3%)	103 (97%)		106.00
		Mean	3 (3%)	104.5 (99%)	1 (1%)	

Table 57. The completeness level in each of the Existence groups among the EPs of the COPD GL

	KRS	Editor	KRs (In presentage) Exist in GS & not in Markup	KRs (In presentage) Exist in GS and in Markup	KRs (In presentage) Not Exist in GS& Exist in Markup	Total
Context	Actors	EP8	11 (92%)	1 (8%)		12.00
		EP5	2 (17%)	10 (83%)		12.00
	Clinical Context	EP8	11 (92%)	1 (8%)		12.00
		EP5	2 (17%)	10 (83%)		12.00
	Strength of recomestation	EP8			3 (0%)	
		EP5				
	Level of evidence	EP8			3 (0%)	
		EP5				
	Mean Context	EP8	11 (92%)	1 (8%)		
		EP5	2 (17%)	10 (83%)		
		Mean	6.5 (54%)	5.5 (46%)		
Intentions	Intermediate Process Intentions	EP8		1 (100%)		1.00
		EP5		1 (100%)		1.00
	Intermediate Outcome Intentions	EP8		1 (100%)		1.00
		EP5		1 (100%)		1.00
	Overall Outcome Intentions	EP8		1 (100%)		1.00
		EP5		1 (100%)		1.00
	Mean Intentions	EP8		1 (100%)		
		EP5		1 (100%)		
		Mean		1 (100%)		
Conditions	Filter Condition	EP8	1 (50%)	1 (50%)		2.00
		EP5		2 (100%)		2.00
	Abort Condition	EP8	3 (60%)	2 (40%)		5.00
		EP5	2 (40%)	3 (60%)		5.00
	Complete Condition	EP8	2 (67%)	1 (33%)		3.00
		EP5		3 (100%)		3.00
	Suspend Condition	EP8	1 (0%)	0 (0%)		1.00
		EP5		1 (100%)		1.00
	Mean Conditions	EP8	1.8 (59%)	1 (31%)		
		EP5	2 (40%)	2.3 (90%)		
		Mean	1.9 (49%)	1.6 (60%)		
PlanBody	PlanBody	EP8		59 (100%)	1 (2%)	59.00
		EP5	4 (7%)	55 (93%)	2 (3%)	59.00
		Mean	2 (3%)	57 (97%)	2 (3%)	

Table 58. The completeness level in each of the Existence groups among the EPs of the HYpoThyrd GL

	KRS	Editor	KRs (In presentage) Exist in GS & not in Markup	KRs (In presentage) Exist in GS and in Markup	KRs (In presentage) Not Exist in GS& Exist in Markup	Total
Context	Actors	EP8		1 (100%)		1.00
		EP5		1 (100%)		1.00
	Clinical Context	EP8		1 (100%)		1.00
		EP5		1 (100%)		1.00
	Mean Context	EP8		1 (100%)		
		EP5		1 (100%)		
		Mean		1 (100%)		
Intentions	Intermediate Process Intentions	EP8		3 (100%)		3.00
		EP5		3 (100%)		3.00
	Intermediate Outcome Intentions	EP8		2 (100%)		2.00
		EP5		2 (100%)		2.00
	Overall Process Intentions	EP8		3 (100%)		3.00
		EP5		3 (100%)		3.00
	Overall Outcome Intentions	EP8		3 (100%)		3.00
		EP5		3 (100%)		3.00
	Mean Intentions	EP8		2.8 (100%)		
		EP5		3 (100%)		
		Mean		2.9 (100%)		
Conditions	Filter Condition	EP8	1 (100%)	0 (0%)		1.00
		EP5		1 (100%)		1.00
	Abort Condition	EP8	1 (33%)	2 (67%)		3.00
		EP5		3 (100%)		3.00
	Complete Condition	EP8				0.00
		EP5				1.00
	setup Condition	EP8		1.00		1.00
		EP5		1 (100%)		1.00
	Mean Conditions	EP8	1 (67%)	1 (56%)	0 (0%)	
		EP5	0 (0%)	1.7 (100%)		
		Mean	0.5 (33%)	1.3 (78%)		
PlanBody	Mean PlanBody	EP8		28 (90%)		31.00
		EP5	2 (6%)	29 (94%)		31.00
		Mean	2.5 (3%)	28.5 (92%)		

Appendix D.2 - Correctness Results

Table 59. The number of instances the scale of [-1,0,1] in both Asbru and Clinical measures, the MQS and its STD for each KR, overall markup and tasks for the PID GL for the declerative KRs

			PID													
			EP1 - KRS(precentage)						EP2 - KRS(precentage)						Mean	
			-1	0	1	Total	MQS	Std	-1	0	1	Total	MQS	Std	MQS	Std
Context	Actors	clinical			11 (100%)	11	1.00	0.00	1 (9%)		10 (91%)	11	0.82	0.60	0.91	0.43
		Asbru			11 (100%)	11	1.00	0.00	2 (18%)		9 (82%)	11	0.64	0.81	0.82	0.59
		Mean MQS					1.00	0.00					0.73	0.70	0.86	0.51
	Clinical Context	clinical			11 (100%)	11	1.00	0.00	2 (18%)		9 (82%)	11	0.64	0.84	0.82	0.60
		Asbru			11 (100%)	11	1.00	0.00	2 (18%)		9 (82%)	11	0.64	0.84	0.82	0.60
		Mean MQS					1.00	0.00					0.64	0.82	0.82	0.60
	Average MQS Context			22			1.00	0.00		22			0.68	0.75	0.84	0.59
Intentions	Intermediate Process Intentions	clinical			10 (100%)	10	1.00	0.00	2 (18%)	1 (9%)	8 (73%)	11	0.55	0.88	0.76	0.65
		Asbru			10 (100%)	10	1.00	0.00	2 (18%)	1 (9%)	8 (73%)	11	0.55	0.88	0.76	0.65
		Mean MQS					1.00	0.00					0.55	0.86	0.76	0.64
	Overall Outcome Intentions	clinical			4 (100%)	4	1.00	0.00			4 (100%)	4	1.00	0.00	1.00	0.00
		Asbru			4 (100%)	4	1.00	0.00			4 (100%)	4	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Mean MQS Intentions			14			1.00	0.00		15			0.67	0.75	0.83	0.55
Conditions	Filter Condition	clinical	1 (3%)	1 (3%)	27 (93%)	29	0.90	0.16	2 (5%)	11 (28%)	26 (67%)	39	0.62	0.59	0.74	0.53
		Asbru		23 (79%)	6 (21%)	29	0.21	0.43	2 (5%)	32 (82%)	5 (13%)	39	0.08	0.42	0.13	0.43
		Mean MQS					0.55	0.53					0.35	0.58	0.43	0.57
	Abort Condition	clinical			10 (100%)	10	1.00	0.00	1 (10%)			10	0.80	0.67	0.90	0.46
		Asbru		7 (70%)	3 (30%)	10	0.30	0.48		7 (70%)	3 (30%)	10	0.30	0.50	0.30	0.48
		Mean MQS					0.65	0.49					0.55	0.62	0.60	0.55
	Complete Condition	clinical			5 (100%)	5	1.00	0.00	1 (17%)	2 (33%)	3 (50%)	6	0.33	0.82	0.64	0.73
		Asbru			5 (100%)	5	1.00	0.00	1 (17%)	2 (33%)	3 (50%)	6	0.33	0.82	0.64	0.73
		Mean MQS					1.00	0.00					0.33	0.78	0.64	0.70
	Suspend Condition	clinical			1 (100%)	1	1.00	0.00				1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00				1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Restart Condition	clinical			1 (100%)	1	1.00	0.00				1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00				1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Mean MQS Conditions			46			0.64	0.51		57			0.41	0.61	0.51	0.58
Plan Body	Plan Body	clinical	9 (8%)	3 (3%)	95 (89%)	107	0.80	0.57	23 (22%)	5 (5%)	78 (74%)	106	0.52	0.83	0.66	0.73
		Asbru	3 (3%)	9 (8%)	95 (89%)	107	0.86	0.41	8 (8%)	24 (23%)	74 (70%)	106	0.62	0.62	0.74	0.54
		Mean MQS Plan Body					0.83	0.50					0.57	0.74	0.70	0.64
Task	Aplication Mean MQS						0.77	0.52					0.51	0.69	0.64	0.63
	QA Mean MQS						0.87	0.45					0.60	0.73	0.74	0.61
	Total Mean MQS						0.82	0.47					0.54	0.71	0.68	0.62

Table 60. The number of instances the scale of [-1,0,1] in both Asbru and Clinical measures, the MQS and its STD for each KR, overall markup and tasks for the COPD GL for the declerative KRs

			COPD													
			EP8 - KRS(precentage)						EP5 - KRS(precentage)						Mean	
			-1	0	1	Total	MQS	std	-1	0	1	Total	MQS	std	MQS	std
Context	Actors	clinical	11 (92%)	0 (0%)	1 (8%)	12	-0.83	0.58	2 (17%)		10 (83%)	12	0.67	0.78	-0.08	1.02
		Asbru	11 (92%)		1 (8%)	12	-0.83	0.58	2 (17%)		10 (83%)	12	0.67	0.78	-0.08	1.02
		Mean MQS					-0.83	0.56					0.67	0.76	-0.08	1.01
	Clinical Context	clinical	11 (92%)		1 (8%)	12	-0.83	0.60	2 (17%)		10 (83%)	12	0.67	0.78	-0.08	1.02
		Asbru	11 (92%)		1 (8%)	12	-0.83	0.60	2 (17%)		10 (83%)	12	0.67	0.78	-0.08	1.02
		Mean MQS					-0.83	0.59					0.67	0.76	-0.08	1.01
	Mean MQS Context			24			-0.83	0.57		24			0.67	0.75	-0.08	1.00
Intentions	Suspend Condition	clinical			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Intermediate Outcome Intentions	clinical			1 (100%)	1	1.00	0.00		1 (100%)		1	0.00	0.00	0.50	0.71
		Asbru			1 (100%)	1	1.00	0.00		1 (100%)		1	0.00	0.00	0.50	0.71
		Mean MQS					1.00	0.00					0.00	0.00	0.50	0.58
	Overall Outcome Intentions	clinical			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Mean MQS Intentions			3			1.00	0.00		3			0.67	0.52	0.83	0.39
Conditions	Filter Condition	clinical	1 (50%)		1 (50%)	2	0.00	1.41			2 (100%)	2	1.00	0.00	0.50	1.00
		Asbru	1 (50%)		1 (50%)	2	0.00	1.41			2 (100%)	2	1.00	0.00	0.50	1.00
		Mean MQS					0.00	1.15					1.00	0.00	0.50	0.93
	Abort Condition	clinical	3 (60%)		2 (40%)	5	-0.20	1.10	2 (40%)		3 (60%)	5	0.20	1.10	0.00	1.05
		Asbru	3 (60%)		2 (40%)	5	-0.20	1.00	2 (40%)		3 (60%)	5	0.20	1.10	0.00	0.99
		Mean MQS					-0.20	0.99					0.20	1.03	0.00	1.00
	Complete Condition	clinical	2 (67%)		1 (33%)	3	-0.33	1.15			3 (100%)	3	1.00	0.00	0.33	1.03
		Asbru	2 (67%)		1 (33%)	3	-0.33	1.15			3 (100%)	3	1.00	0.00	0.33	1.03
		Mean MQS					-0.33	1.03					1.00	0.00	0.33	0.98
	Suspend Condition	clinical	1 (100%)			1	-1.00	0.00				1	1.00	0.00	0.00	0.00
		Asbru	1 (100%)			1	-1.00	0.00				1	1.00	0.00	0.00	0.00
		Mean MQS					-1.00	0.00					1.00	0.00	0.00	0.00
	Mean MQS Conditions			11			-0.27	0.97		11			0.64	0.79	0.18	0.98
Plan Body	Plan Body	clinical	2 (3%)	3 (5%)	55 (92%)	60	0.88	0.42	11 (18%)	5 (8%)	45 (74%)	61	0.56	0.79	0.72	0.66
		Asbru	3 (5%)	45 (75%)	12 (20%)	60	0.15	0.48	6 (10%)	10 (16%)	45 (74%)	61	0.64	0.67	0.40	0.63
		Mean MQS Plan Body					0.52	0.58					0.60	0.73	0.56	0.66
Task	Aplication Mean MQS						0.39	0.70					0.60	0.73	0.50	0.72
	QA Mean MQS						0.16	0.85					0.62	0.73	0.39	0.83
	Total Mean MQS						0.11	0.86					0.62	0.73	0.37	0.83

Table 61. The number of instances the scale of [-1,0,1] in both Asbru and Clinical measures, the MQS and its STD for each KR, overall markup and tasks for the HypoThyrd GL for the declerative KRs

			HypoThyrd													
			EP8 - KRS(precentage)						EP5 - KRS(precentage)						Mean	
			-1	0	1	Total	MQS	std	-1	0	1	Total	MQS	std	MQS	std
Context	Actors	clinical			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Clinical Context	clinical			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Mean MQS Context				2		1.00	0.00			2		1.00	0.00	1.00	0.00
Intentions	Intermediate Process Intentions	clinical			3 (100%)	3	1.00	0.00			3 (100%)	3	1.00	0.00	1.00	0.00
		Asbru			3 (100%)	3	1.00	0.00			3 (100%)	3	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Intermediate Outcome Intentions	clinical			2 (100%)	2	1.00	0.00			2 (100%)	2	1.00	0.00	1.00	0.00
		Asbru			2 (100%)	2	1.00	0.00			2 (100%)	2	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Overall Process Intentions	clinical			3 (100%)	3	1.00	0.00			3 (100%)	3	1.00	0.00	1.00	0.00
		Asbru			3 (100%)	3	1.00	0.00			3 (100%)	3	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Overall Outcome Intentions	clinical			3 (100%)	3	1.00	0.00			3 (100%)	3	1.00	0.00	1.00	0.00
		Asbru			3 (100%)	3	1.00	0.00			3 (100%)	3	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Mean MQS Intentions				8		1.00	0.00			8		1.00	0.00	1.00	0.00
Conditions	Filter Condition	clinical	1 (0%)			1	-1.00	0.00			1 (100%)	1	1.00	0.00	0.00	0.00
		Asbru	1 (0%)			1	-1.00	0.00			1 (100%)	1	1.00	0.00	0.00	0.00
		Mean MQS					-1.00	0.00					1.00	0.00	0.00	0.00
	SetUP Condition	clinical			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Asbru			1 (100%)	1	1.00	0.00			1 (100%)	1	1.00	0.00	1.00	0.00
		Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
	Abort Condition	clinical	1 (33%)			3	0.33	1.15			3 (100%)	3	1.00	0.00	0.67	0.82
		Asbru	1 (33%)			3	0.33	1.15			3 (100%)	3	1.00	0.00	0.67	0.82
		Mean MQS					0.33	1.03					1.00	0.00	0.67	0.78
	complete Condition	clinical							1 (0%)			1	-1.00	0.00	-1.00	1.00
		Asbru							1 (0%)			1	-1.00	0.00	-1.00	0.67
		Mean MQS											-1.00	0.00	-1.00	0.00
	Mean MQS Conditions		5				0.20	1.03			6		0.67	0.00	0.43	0.91
Plan Body	Plan Body	clinical	3 (10%)	1 (3%)	27 (87%)	31	0.77	0.68	3 (10%)		28 (90%)	31	0.81	0.60	0.79	0.64
		Asbru	3 (10%)	9 (29%)	19 (61%)	31	0.52	0.68	2 (6%)	1 (3%)	28 (90%)	31	0.84	0.53	0.68	0.63
		Mean MQS Plan Body					0.65	0.68					0.82	0.56	0.73	0.63
Task	Aplication Mean MQS						0.58	0.74					0.80	0.60	0.69	0.63
	QA Mean MQS						0.73	0.66					0.87	0.52	0.80	0.60
	Total Mean MQS						0.67	0.67					0.84	0.52	0.76	0.60

Table 62. The number of instances the scale of [-1,0,1] in both Asbru and Clinical measures, the MQS and its STD for each KR for the PID GL for the procedural KRs

		PID													
		EP1 - KRS(precentage)						EP2 - KRS(precentage)						Mean	
		-1.00	0.00	1.00	Total	MQS	STD	-1.00	0.00	1.00	Total	MQS	STD	MQS	STD
Cyclic	clinical	3 (12%)	3 (12%)	19 (76%)	25.00	0.64	0.70	16 (67%)	3 (13%)	5 (21%)	24.00	-0.46	0.83	0.09	0.95
	Asbru	1 (4%)	4 (16%)	20 (80%)	25.00	0.76	0.51	5 (21%)	17 (71%)	2 (8%)	24.00	-0.13	0.54	0.32	0.69
	Mean MQS					0.70	0.61					-0.29	0.71	0.21	0.84
If Then Else	clinical			16 (100%)	16.00	1.00	0.00	1 (6%)			16.00	0.88	0.50	0.94	0.35
	Asbru		1 (6%)	15 (94%)	16.00	0.94	0.25	1 (0%)	1 (6%)	14 (88%)	16.00	0.81	0.54	0.88	0.42
	Mean MQS					0.97	0.18					0.84	0.51	0.91	0.39
Plan Activation	clinical			9 (100%)	9.00	1.00	0.00	1 (11%)		8 (89%)	9.00	0.78	0.67	0.89	0.47
	Asbru			9 (100%)	9.00	1.00	0.00	1 (11%)		8 (89%)	9.00	0.78	0.67	0.89	0.47
	Mean MQS					1.00	0.00					0.78	0.65	0.89	0.46
Parallel Plan	clinical	2 (18%)		9 (82%)	11.00	0.64	0.81	4 (36%)		7 (64%)	11.00	0.27	1.01	0.45	0.91
	Asbru		2 (18%)	9 (82%)	11.00	0.82	0.39		4 (36%)	7 (64%)	11.00	0.64	0.50	0.73	0.45
	Mean MQS					0.73	0.62					0.45	0.80	0.59	0.72
Sequential Plan	clinical	2 (14%)		12 (86%)	14.00	0.71	0.73	1 (7%)	2 (14%)	11 (79%)	14.00	0.71	0.61	0.71	0.66
	Asbru	1 (7%)	1 (7%)	12 (86%)	14.00	0.79	0.58	1 (7%)	2 (14%)	11 (79%)	14.00	0.71	0.61	0.75	0.59
	Mean MQS					0.75	0.65					0.71	0.61	0.73	0.62
Simple Plan	clinical	2 (7%)		26 (93%)	28.00	0.86	0.52			28 (100%)	28.00	1.00	0.00	0.93	0.37
	Asbru	1 (4%)	1 (4%)	26 (93%)	28.00	0.89	0.42			28 (100%)	28.00	1.00	0.00	0.95	0.30
	Mean MQS					0.88	0.47					1.00	0.00	0.94	0.34
To be Defined	clinical			4 (100%)	4.00	1.00	0.00			4 (100%)	4.00	1.00	0.00	1.00	0.00
	Asbru			4 (100%)	4.00	1.00	0.00			4 (100%)	4.00	1.00	0.00	1.00	0.00
	Mean MQS					1.00	0.00					1.00	0.00	1.00	0.00
Total Mean MQS						0.83	0.50					0.57	0.74	0.70	0.64

Table 63. The number of instances the scale of [-1,0,1] in both Asbru and Clinical measures, the MQS and its STD for each KR for the COPD GL for the procedural KRs

		COPD													
		EP8 - KRS(precentage)						EP5 - KRS(precentage)						Mean	
		-1.00	0.00	1.00	Total	MQS	STD	-1.00	0.00	1.00	Total	MQS	STD	MQS	STD
Cyclic	clinical			2 (100%)	2.00	1.00	0.00	1 (50%)	1 (50%)		2.00	-0.50	0.71	0.25	0.96
	Asbru		1 (50%)	1 (50%)	2.00	0.50	0.71	1 (50%)	1 (50%)		2.00	-0.50	0.71	0.00	0.82
	Mean MQS	2.00				0.75	0.50	2.00				-0.50	0.58	0.13	0.83
If Then Else	clinical		2 (50%)	2 (50%)	4.00	0.50	0.58	1 (25%)		3 (75%)	4.00	0.50	1.00	0.50	0.76
	Asbru	2 (50%)	1 (25%)	1 (25%)	4.00	-0.25	0.96	1 (25%)		3 (75%)	4.00	0.50	1.00	0.13	0.99
	Mean MQS	4.00				0.13	0.83	4.00				0.50	0.93	0.31	0.87
Parallel Plan	clinical	2 (25%)		6 (75%)	8.00	0.50	0.93	3 (38%)	4 (50%)	1 (13%)	8.00	-0.25	0.71	0.13	0.89
	Asbru		7 (88%)	1 (13%)	8.00	0.13	0.35		7 (88%)	1 (13%)	8.00	0.13	0.35	0.13	0.34
	Mean MQS	8.00				0.31	0.70	8.00				-0.06	0.57	0.13	0.66
Sequential Plan	clinical			2 (67%)	3.00	0.67	0.58	3 (100%)			3.00	-1.00	0.00	-0.17	0.98
	Asbru	1 (33%)	2 (67%)		3.00	-0.33	0.58	1 (33%)	2 (67%)		3.00	-0.33	0.58	-0.33	0.98
	Mean MQS	3.00				0.17	0.75	3.00				-0.67	0.52	-0.25	0.75
Simple Plan	clinical			39 (100%)	39.00	1.00	0.00	3 (8%)		35 (92%)	38.00	0.84	0.55	0.92	0.39
	Asbru		34 (87%)	5 (13%)	39.00	0.13	0.34	3 (8%)		35 (92%)	38.00	0.84	0.55	0.49	0.49
	Mean MQS	39.00				0.56	0.50	38.00				0.84	0.54	0.70	0.39
To be Defined	clinical			4 (100%)	4.00	1.00	0.00			6 (100%)	6.00	1.00	0.00	1.00	0.00
	Asbru			4 (100%)	4.00	1.00	0.00			6 (100%)	6.00	1.00	0.00	1.00	0.00
	Mean MQS	4.00				1.00	0.00	6.00				1.00	0.00	1.00	0.00
Total Mean MQS						0.52	0.58					0.60	0.73	0.56	0.66

Table 64. The number of instances the scale of [-1,0,1] in both Asbru and Clinical measures, the MQS and its STD for each KR for the hypoThyrd GL for the procedural KRs

		COPD													
		EP8 - KRS(precentage)						EP5 - KRS(precentage)						Mean	
		-1.00	0.00	1.00	Total	MQS	STD	-1.00	0.00	1.00	Total	MQS	STD	MQS	STD
Cyclic	clinical			2 (100%)	2.00	1.00	0.00	1 (50%)	1 (50%)		2.00	-0.50	0.71	0.25	0.96
	Asbru		1 (50%)	1 (50%)	2.00	0.50	0.71	1 (50%)	1 (50%)		2.00	-0.50	0.71	0.00	0.82
	Mean MQS	2.00				0.75	0.50	2.00				-0.50	0.58	0.13	0.83
If Then Else	clinical		2 (50%)	2 (50%)	4.00	0.50	0.58	1 (25%)		3 (75%)	4.00	0.50	1.00	0.50	0.76
	Asbru	2 (50%)	1 (25%)	1 (25%)	4.00	-0.25	0.96	1 (25%)		3 (75%)	4.00	0.50	1.00	0.13	0.99
	Mean MQS	4.00				0.13	0.83	4.00				0.50	0.93	0.31	0.87
Parallel Plan	clinical	2 (25%)		6 (75%)	8.00	0.50	0.93	3 (38%)	4 (50%)	1 (13%)	8.00	-0.25	0.71	0.13	0.89
	Asbru		7 (88%)	1 (13%)	8.00	0.13	0.35		7 (88%)	1 (13%)	8.00	0.13	0.35	0.13	0.34
	Mean MQS	8.00				0.31	0.70	8.00				-0.06	0.57	0.13	0.66
Sequential Plan	clinical			2 (67%)	3.00	0.67	0.58	3 (100%)			3.00	-1.00	0.00	-0.17	0.98
	Asbru	1 (33%)	2 (67%)		3.00	-0.33	0.58	1 (33%)	2 (67%)		3.00	-0.33	0.58	-0.33	0.98
	Mean MQS	3.00				0.17	0.75	3.00				-0.67	0.52	-0.25	0.75
Simple Plan	clinical			39 (100%)	39.00	1.00	0.00	3 (8%)		35 (92%)	38.00	0.84	0.55	0.92	0.39
	Asbru		34 (87%)	5 (13%)	39.00	0.13	0.34	3 (8%)		35 (92%)	38.00	0.84	0.55	0.49	0.49
	Mean MQS	39.00				0.56	0.50	38.00				0.84	0.54	0.70	0.39
To be Defined	clinical			4 (100%)	4.00	1.00	0.00			6 (100%)	6.00	1.00	0.00	1.00	0.00
	Asbru			4 (100%)	4.00	1.00	0.00			6 (100%)	6.00	1.00	0.00	1.00	0.00
	Mean MQS	4.00				1.00	0.00	6.00				1.00	0.00	1.00	0.00
Total Mean MQS						0.52	0.58					0.60	0.73	0.56	0.66

Appendix D.3 – Proportion of scores in each KR

Table 65. The proportions of scores in the Context KR class

	Actors						Clinical Context					
	Asbru		Clinic		all		Asbru		Clinic		all	
-1	15	15.63%	14	14.58%	29	30.21%	14	15.22%	14	15.22%	28	30.43%
0	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%	0	0.00%
1	33	34.38%	34	35.42%	67	69.79%	32	34.78%	32	34.78%	64	69.57%

Table 66. The proportions of scores in the Intentions KR class

	Intermediate Process Intentions						Overall Outcome Intentions					
	Asbru		Clinic		all		Asbru		Clinic		all	
-1	2	3.70%	2	3.70%	4	7.41%	0	0.00%	0	0.00%	0	0.00%
0	1	1.85%	1	1.85%	2	3.70%	0	0.00%	0	0.00%	0	0.00%
1	24	44.44%	24	44.44%	48	88.89%	16	50.00%	16	50.00%	32	100.00%

Table 67. The proportions of scores in the Conditions KR class

	Abort Condition						Complete Condition						FIlter Condition					
	Asbru		Clinic		all		Asbru		Clinic		all		Asbru		Clinic		all	
-1	5	7.14%	7	10.00%	12	17.14%	4	12.50%	4	12.50%	8	25.00%	4	2.67%	5	3.33%	9	6.00%
0	14	20.00%	0	0.00%	14	20.00%	2	6.25%	2	6.25%	4	12.50%	55	36.67%	12	8.00%	67	44.67%
1	16	22.86%	28	40.00%	44	62.86%	10	31.25%	10	31.25%	20	62.50%	16	10.67%	58	38.67%	74	49.33%

Table 68. The proportions of scores in the Plan-Body(I) KR class

	cyclic						IfThenElse						parallelPlan					
	Asbru		Clinic		all		Asbru		Clinic		all		Asbru		Clinic		all	
-1	7	6.25%	20	17.86%	27	24.11%	4	3.85%	2	1.92%	6	5.77%	0	0.00%	11	12.94%	11	12.94%
0	22	19.64%	6	5.36%	28	25.00%	9	8.65%	2	1.92%	11	10.58%	21	24.71%	4	4.71%	25	29.41%
1	27	24.11%	30	26.79%	57	50.89%	39	37.50%	48	46.15%	87	83.65%	22	25.88%	27	31.76%	49	57.65%

Table 69 The proportions of scores in the Plan-Body(II) KR class

	planactiv						sequentialPlan						simpleAction						TBD					
	Asbru		Clinic		all		Asbru		Clinic		all		Asbru		Clinic		all		Asbru		Clinic		all	
-1	1	2.08%	1	2.08%	2	4.17%	6	7.59%	8	10.13%	14	17.72%	7	2.22%	9	2.86%	16	5.08%	0	0.00%	0	0.00%	0	0.00%
0	0	0.00%	0	0.00%	0	0.00%	8	10.13%	3	3.80%	11	13.92%	36	11.43%	0	0.00%	36	11.43%	0	0.00%	0	0.00%	0	0.00%
1	23	47.92%	23	47.92%	46	95.83%	26	32.91%	28	35.44%	54	68.35%	114	36.19%	149	47.30%	263	83.49%	16	50.00%	16	50.00%	32	100.00%

Table 70 The proportions of scores aggregated by classes

	Context						Intentions						Conditions						plan- body					
	Asbru		Clinic		all		Asbru		Clinic		all		Asbru		Clinic		all		Asbru		Clinic		all	
-1	29	15.43%	28	14.89%	57	30.32%	2	2.33%	2	2.33%	4	4.65%	13	5.16%	16	6.35%	29	11.51%	25	3.23%	51	6.58%	76	9.81%
0	0	0.00%	0	0.00%	0	0.00%	1	1.16%	1	1.16%	2	2.33%	71	28.17%	14	5.56%	85	33.73%	96	6.51%	15	1.94%	111	14.32%
1	65	34.57%	66	35.11%	131	69.68%	40	46.51%	40	46.51%	80	93.02%	42	16.67%	96	38.10%	138	54.76%	267	19.59%	321	41.42%	588	75.87%

Table 71. The proportions of scores for all classes

	All classes					
	Asbru		Clinic		all	
-1	69	5.30%	97	7.46%	166	12.76%
0	168	12.91%	30	2.31%	198	15.22%
1	414	31.82%	523	40.20%	937	72.02%

Appendix D.4 – Proportion test Results

Table 72. The proportion test results between the Asbru and Clinical measure in each KR class The Z value, the P value, and the confidence interval of the test is also showed

	ABRU		CLINIC		z statstic	P value	Confidence Interval (95%)
	Proportion	N	Proportion	N			
Context	34.57%	65	35.11%	66	0.11 *	0.914	[-0.102 , 0.091]
Intentions	46.51%	40	46.51%	40	0 *	1.000	[-0.149 , 0.149]
Conditions	16.67%	42	38.10%	96	5.39	0.000	[-0.29 , -0.139]
Planbody	34.45%	267	41.42%	321	2.83	0.005	[-0.118 , -0.021]
All Classes	31.82%	414	40.20%	523	4.45	0.000	[-0.121 , -0.047]

*: The non significant (P>0.05) Z values which are in the acceptances interval, that is with no significant different in proportion

Table 73 The proportion test results between the Asbru and Clinical measure in each KR Type The Z value, the P value, and the confidence interval of the test is also showed

	ABRU		CLINIC		z statstic	P value	Confidence Interval (95%)
	Proportion	N	Proportion	N			
Actors	34.38%	33	35.42%	34	0.15 *	0.880	[-0.145 , 0.124]
Clinical Context	34.78%	32	34.78%	32	0 *	1.000	[-0.138 , 0.138]
Intermediate Process Intentions	44.44%	24	44.44%	24	0 *	1.000	[-0.187 , 0.187]
Overall Outcome Intentions	50.00%	16	50.00%	16	0 *	1.000	[-0.245 , 0.245]
Abort Condition	22.86%	16	40.00%	28	2.18	0.029	[-0.323 , -0.02]
Complete Condition	31.25%	10	31.25%	10	0 *	1.000	[-0.227 , 0.227]
FIlter Condition	10.67%	16	38.67%	58	5.63	0.000	[-0.372 , -0.188]
cyclic	24.11%	27	26.79%	30	0.46 *	0.645	[-0.141 , 0.087]
IfThenElse	37.50%	39	46.15%	48	1.27 *	0.206	[-0.22 , 0.047]
parallelPlan	25.88%	22	31.76%	27	0.85 *	0.397	[-0.195 , 0.077]
planactiv	0.479166667	23	47.92%	23	0 *	1.000	[-0.2 , 0.2]
sequentialPlan	32.91%	26	35.44%	28	0.34 *	0.737	[-0.173 , 0.123]
simpleAction	36.19%	114	47.30%	149	2.83	0.005	[-0.188 , -0.035]
TBD	50.00%	16	50.00%	16	0 *	1.000	[-0.245 , 0.245]

Appendix D.5 – Error Types Results

Table 74 The error types in the PID GL

		PID								
		ASBRU				CLINIC				Total
		Asbru semantics content not accurate	Asbru semantics content not well specified	No reference to guideline knowledge	Erorrs Asbru	Clinical content not accurate	Clinical content not complete	Clinical semantics not well specified	Erorrs clinical	
Simple Action	EP1		1 (5%)		1 (5%)	2 (9%)	19 (86%)		21 (95%)	22
	EP2						25 (100%)		25 (100%)	25
	Sum		1 (2%)		1 (2%)	2 (4%)	44 (94%)		46 (98%)	47
Plan activation	EP1									
	EP2									
	Sum									
Cyclical plan	EP1		4 (44%)		4 (44%)	3 (33%)			5 (56%)	9
	EP2	12 (22%)	19 (35%)		31 (57%)	14 (26%)	8 (15%)	1 (2%)	23 (43%)	54
	Sum	12 (19%)	23 (37%)		35 (56%)	17 (27%)	10 (16%)	1 (2%)	28 (44%)	63
if-then-else	EP1			1 (100%)	1 (100%)					1
	EP2									
	Sum			1 (100%)	1 (100%)					1
Subplans – parallel order	EP1	1 (25%)	1 (25%)		2 (50%)	2 (50%)			2 (50%)	4
	EP2	1 (17%)	2 (33%)		3 (50%)	3 (50%)			3 (50%)	6
	Sum	2 (20%)	3 (30%)		5 (50%)	5 (50%)			5 (50%)	10
Subplans – sequential order	EP1		2 (67%)		2 (67%)	1 (33%)			1 (33%)	3
	EP2		2 (67%)		2 (67%)	1 (33%)			1 (33%)	3
	Sum		4 (67%)		4 (67%)	2 (33%)			2 (33%)	6
Total		14 (11%)	31 (24%)	1 (1%)	46 (36%)	26 (20%)	54 (43%)	1 (1%)	81 (64%)	127

Table 75 The error types in the COPD GL

		COPD							
		ASBRU			CLINICAL				Total
		Asbru semantics content not well specified	No reference to guideline knowledge	Total Erorrs Asbru	Clinical content not accurate	Clinical content not complete	Clinical semantics not well specified	Total Erorrs clinical	
Simple Action	EP8	33 (100%)		33 (100%)					33
	EP5								
	Sum	33 (100%)		33 (100%)					33
Plan activation	EP8								
	EP5								
	Sum								
Cyclical plan	EP8	1 (33%)		2 (67%)			1 (33%)	1 (33%)	3
	EP5	2 (67%)		2 (4%)	1 (2%)			1 (2%)	3
	Sum	3 (50%)		4 (67%)	1 (2%)		1 (17%)	2 (33%)	6
if-then-else	EP8	2 (33%)	2 (33%)	4 (67%)		1 (17%)	1 (17%)	2 (33%)	6
	EP5								
	Sum	2 (33%)	2 (33%)	4 (400%)		1 (17%)	1 (17%)	2 (33%)	6
Subplans – parallel order	EP8	7 (78%)		7 (78%)	1 (25%)		1 (11%)	2 (22%)	9
	EP5	8 (50%)		8 (133%)	8 (133%)			8 (133%)	16
	Sum	15 (60%)		15 (60%)	9 (90%)		1 (4%)	10 (40%)	25
Subplans – sequential order	EP8	2 (67%)		2 (67%)	1 (33%)			1 (33%)	3
	EP5	3 (50%)		3 (100%)	3 (100%)			3 (100%)	6
	Sum	5 (56%)		5 (56%)	4 (67%)			4 (44%)	9
Total		58 (73%)	3 (2%)	61 (77%)	14 (18%)	1 (1%)	3 (4%)	18 (14%)	79

Table 76 The error types in the HypoThyrd GL

		HypoThyrd					
		ASBRU		CLINIC			Total
		Asbru semantics content not well specified	total asbru	total asbru	Clinical content not accurate	Clinical semantics not well specified	
Simple Action	EP8	2 (100%)	2 (100%)				2
	EP5						
	Sum	2 (100%)	2 (100%)				2
Plan activation	EP8						
	EP5						
	Sum						
Cyclical plan	EP8						
	EP5						
	Sum						
if-then-else	EP8	4 (100%)	4 (100%)				4
	EP5						
	Sum	4 (100%)	4 (100%)				4
Subplans – parallel order	EP8	1 (100%)	1 (100%)				1
	EP5						
	Sum	1 (100%)	1 (100%)				1
Subplans – sequential order	EP8	1 (50%)	1 (50%)	1 (50%)		1 (50%)	2
	EP5	1 (33%)	1 (33%)	1 (33%)	1 (33%)	2 (67%)	3
	Sum	2 (40%)	2 (40%)	2 (40%)	1 (20%)	3 (60%)	5
Total		9 (75%)	9 (75%)	2 (17%)	1 (8%)	3 (25%)	12

תקציר

רקע: במסגרת המעבדה למערכות מידע רפואיות פותחה ספרייה דיגיטאלית מבוססת אינטרנט של קמ״רים בשם ״דגל״- **DeGeL** (Digital Electronic Guideline Library) הכוללת סט של כלים, ביניהם **URUZ** - כלי מבוסס אינטרנט לרכישת הידע הרפואי הרב הטמון בקו מנחה רפואי (קמ״ר). במסגרת המחקר, פותחה והוערכה מתודולוגיה לרכישת ידע *תהליכי והצהרתי* הטמון בקמ״ר. מתודולוגיה זו כוללת הגדרת והבניית הידע הטמון בקמ״ר ע״י רופאים, מייצוגו הטקסטואלי, דרך ייצוג חצי- מובנה ועד לייצוג מובנה המאפשר הרצתו ע״י מחשב בשפת ASBRU .

מטרת המחקר: פיתוח מתודולוגיה לתהליך רכישת הידע ע״י רופאים תוך שימוש בכלים להגדרה והבנייה של קמ״רים במספר רמות ייצוג, והערכה של תהליך זה במדדים כמותיים ואיכותיים

שיטות המחקר:לצורך המחקר, נבחרו שלושה קמר״ים משלוש דיסציפלינות רפואיות שונות ששימשו כמקור הטקסטואלי לרכישת הידע ע״י הרופאים: דלקת באברי המין של האישה (*Pelvic inflammatory disease*), חסימה כרונית בדרכי הנשימה (*Chronic Obstructive Pulmonary Disease*), ותת -תריסיות (*hypothyroidism*).

בשלב הראשון נוצר מסמך ״קונצנזוס תלוי שפת הבנייה״ שהוא מסמך מובנה המתאר בצורה סכמאתית פרשנות של הקמ״ר אשר מוסכמת ע״י הרופא ומהנדס הידע, וכולל את ההוראות הקליניות הרפואיות של הקמ״ר ואת הלוגיקה הסמנטית של שפת ההבניה. לאחר למידת שפת ההבניה - ASBRU וסביבת העבודה (DeGeL) ואימון בכלי לרכישת הידע ה- URUZ, ערך כל רופא לבד (לעיתים בסביבת עבודתו) תוך שימוש ב URUZ , בקונצנזוס ובידע האישי שלו, מסמך חצי-מובנה בשפת ASBRU שייצג את הקמ״ר (לכל קמ״ר הוכנו שני מסמכים בידי שני רופאים שונים - סה״כ שישה מסמכים חצי-מובנים).

לשם הערכה של המסמכים החצי-מובנים של כל רופא ,הוגדרה לכל קמ״ר אמת מידה מיטבית: מסמך חצי-מובנה שנעשה ע״י רופא ומהנדס ידע המתאר את ההבניה הטובה ביותר של הקמ״ר. כל מסמך חצי-מובנה שנוצר ע״י הרופאים שערכו את הידע הושווה מול אמת מידה זו. בנוסף, הוגדרו מדדים סובייקטיביים ואובייקטיבים להערכה איכותית וכמותית של הידע שנרכש בכל מסמך חצי-מובנה של כל רופא: המדדים הסובייקטיביים כללו שאלונים לגבי רמת הידע של הרופאים בסביבת העבודה , שפת ההבניה והשימושיות ב- URUZ. המדדים האובייקטיבים התייחסו ל 2 קטגוריות עיקריות: **שלמות** הידע הנרכש, כלומר כמה תוכן מאמת המידה קיים במסמך החצי-מובנה של הרופאים (כגון תתי תכניות שהוגדרו מראש), **ונכונות** הידע כלומר כמה הידע שייצגו הרופאים נכון מבחינה 1)רפואית קלינית 2) סמנטית, לפי שפת ASBRU. המדדים נבדקו בכמה חתכים מעניינים: לכל מרכיב בשפת ASBRU, לכל מסמך חצי-מובנה שרופא יצר, לכל קמ״ר ולכל הקמרי״ם יחד. תהליך ההערכה בוצע ע״י מהנדס ידע ורופא יחד,ונעשה תוך שימוש בכלי גראפי שפותח לצורך זה, ומאפשר הערכה של המסמכים החצי-מובנים באופן מבוזר.

תוצאות המחקר: לא היה הבדל מובהק באיכות העריכה ע״י רופאים של מרכיבי הידע השונים. שלמות הייצוג של כל הרופאים עבור כל מרכיבי הידע של כל הקמר״ים הייתה בממוצע מעל 96% לכל הקמרי״ם. מבחינת נכונות הייצוג, הייתה שונות ניכרת בין הרופאים (ממוצע של 0.6 בסקלה שבין 1- ל 1) , אך בסה״כ ההבניה הייתה בדרך כלל באיכות טובה. השונות בין הרופאים באה לידי ביטוי בסוגי השגיאות הקליניות וסמנטיות שנעשו, שהיו שונות באופיין בין הקמר״ים.

מסקנות המחקר: בהינתן קונצנזוס תלוי שפת הבניה וייצוג טקסטואלי של הקמ״ר, רופא יכול לייצג באופן חצי מובנה את הידע הטמון בקמ״ר באופן שלם ונכון.

מחקר עתידי צריך לכלול כלים עם אוריינטציה יותר גראפית. ממשק ראשוני כזה כבר פותח במסגרת המחקר לאור המסקנות הראשוניות

אוניברסיטת בן-גוריון בנגב
הפקולטה למדעי ההנדסה
המחלקה להנדסת מערכות מידע

הערכה של מתודולוגיה לרכישת ידע של קוים מנחים רפואיים במספר רמות ייצוג

חיבור זה מהווה חלק מהדרישות לקבלת תואר מגיסטר בהנדסה

מאת: ארז שלום

מנחים: פרופ' יובל שחר
דר' מירב טייב-מימון

חתימת המחבר..... תאריך.....

אישור המנחה/ים..... תאריך 5.3.06

..... תאריך 29.03.06

אישור יו"ר ועדת תואר שני מחלקתית..... תאריך 5.3.06

Prof. Yuval Shahar, M.D., Ph.D.
Head, Dept. of Info... Systems Eng.
Head, Gradua... Committee
Head, Medical Informatics Research Center
Ben-Gurion University of the Negev
Beer-Sheva, Israel

אוניברסיטת בן-גוריון בנגב
הפקולטה למדעי ההנדסה
המחלקה להנדסת מערכות מידע

הערכה של מתודולוגיה לרכישת ידע של קוים מנחים רפואיים במספר רמות ייצוג

חיבור זה מהווה חלק מהדרישות לקבלת תואר מגיסטר בהנדסה

מאת: ארז שלום

March 2006 **אדר, תשס"ו**