A Guideline Engine For Knowledge Management in Clinical Decision Support Systems (CDSSs)

Michele Ceccarelli[a], Alessandro De Stasio[a], Antonio Donatiello[b], Dante Vitale[b]
[a] *University Of Sannio, RCOST (Research Centre On Software Technology), Benevento, 82100 Italy*
[b] *Unlimited Software S.r.l., Napoli, 80143 Italy*
e-mail: ceccarelli@unisannio.it, a.destasio@u-s.it, a.donatiello@u-s.it, dv@unlimitedsoftware.it

Abstract

The application of scientific methodology to clinical practice is typically realized through recommendations, policies and protocols represented as Clinical Practice Guidelines (CPGs). CPGs have the purpose to help the clinicians in their choices and to improve the patient care process. Currently, there have been considerable efforts in digital CPGs for their application to build Clinical Decision Support Systems (CDSSs) in order to deploy them in several hospitals.

The representation of guidelines and their introduction in Clinical Information System (CIS) can lead to efficient Clinical Decision Support Systems (CDSS), however this poses several interesting challenges as it involves problems of knowledge representation, inference, workflow definition, access to unstructured databases of medical records and others.

In this paper we describe the architecture of the Guideline Engine, as part of the KON^3 (Knowledge ON ONcology through ONtology) project. We use a semantic web approach – employing a domain ontology, a patient ontology, decision rules and a Guideline Engine formed by a Process Engine and by a Rule Engine. A Guideline Engine is a computer program which can interpret a clinical guideline represented in a computerized format and perform actions towards the user of an electronic health record (EHR). We also report a specific case study of the application of the model in oncology.

Index Terms *- CDSS (Clinical Decision Support System), Ontology, CPGs (Clinical Practice Guidelines), Protégé, KON^3 (Knowledge ON ONcology through ONtology), SAGE, Process Engine, Rule Engine, EBM (Evidence Based Medicine)*

1. Introduction

Clinical Practice Guidelines (CPGs) are disease-specific recommendations to assist clinical decision-making in accordance with best evidence.

Several studies have demonstrated that health professionals show more effective compliance with guidelines when they are embedded in the knowledge layer of the CDSS and provide a customized management protocol for the individual patient at the point of care [13]. However, the sustainable and successful application of CPG requires a sequence of activities, in particular the CPG needs to be formally computerized to enable it to be executed by computer systems, and to be systematically incorporated within a CDSS.

CPG guided DSS [7][14][17] are particularly useful in clinical settings where health different clinicians are required to deal with complex or unusual cases with the aim of standardize and share the knowledge about clinical treatments. In such situations, CPG-based DSS can guide the clinician's actions and suggest proper recommendations.

In order to achieve a CPG-guided CDSS is necessary to (a) encode CPGs in a structured format at semantic level (ontology); (b) transform the CPGs inherent decision logic into medically salient decision rules; (c) execute the computerized CPG to achieve decision support; (d) ensure the validity of the transformed knowledge and to provide trust in the recommended actions.

In literature, there are several theoretical approaches in CPGs systems, see for example [1] and [21]. On the contrary there are very few papers describing experiences about the realization of a CDSS that reports approaches, architectures and implementation details including issues, limitations and assumptions.

SAGE Project (Standard-based Sharable Active Guideline Environment) [24][25], involves a Guideline Model, a Protégé based tool for encoding, viewing and testing CPGs, the knowledge deployment process and the knowledge execution architecture. However no detail information about the architecture and functionalities of its Execution Engine is available in literature, except [20].

Another interesting work is reported in [12] suggesting an approach and architecture for implementing a CDSS that adopts SAGE Guideline Model. The followed approach involves the translation of the guideline representations to a proprietary knowledge model and to store them in knowledge repository.

In order to be integrated into the clinical workflow, the guideline based approach should depend on the specific patient and pathology at the hand. This implies that CDSS should be integral part of the CIS and the inference engine must be linked with all available clinical records of the patient [6]. Indeed, the wide-spread distribution and use of computable CPG content can be improved if the research community focuses on lack of standards for representing medical knowledge, and on the prohibitive complexity and expense required to adapt encoded guideline content across the heterogeneity of data structures, semantics, and medical vocabularies in use in the nation's health care information systems.

The main objective of the KON^3 [4], a joint effort between

companies, university and regional government agencies, is to obtain a sharable knowledge based on CPG at a reasonable cost and in a form that can be integrated into the clinician's workflow. The main features of KON^3 is the adoption an Ontology for representation of guidelines in addition to a registry based healthcare information infrastructure.

Here we describe the main architecture and functionalities of the central element of developed system consisting into a Guideline Engine together with its application within an oncology environment.

KON^3 CDSS is a generic system, because it's guideline independent provided that it's conforms to adopted Guideline Model. If this requirement is met, so it's possible to encode end execute any guideline.

The paper is organized as follows, *Section II* describes KON^3 project and the adopted architecture. *Section III* describes the KON^3 underlying knowledge representation while in *Section IV*, the architecture of the Guideline Engine is described in details. *Section V* describes a case of study in oncology environment, in particular in the Breast Cancer environment. This guideline, designed and represented at ontology level, is executed by the KON^3 Guideline Engine. *Section VI* contains conclusions and future works.

2. KON^3

KON^3 architecture is described in this section. As shown in Figure 1, the KON^3 CDSS is composed by:

- **Knowledge Base**, it's the knowledge representation at semantic level. It's divided into:
  - *Guideline*, it represents the guideline model;
  - *VMR (Virtual Medical Record)*, it represents the patient data;
  - *Vocabulary*, it contains information about the used vocabulary;
  - *Expression*, it is the module to represent the rules.
- **Guideline Engine**, it is the module to guideline executions. It's divided into:
  - *Process Engine*, it's the scheduler of the guideline;
  - *Rule Engine*, it is the component that executes the rules.
- **Guideline Editor**, it provides the necessary tools to design a guideline.

The CDSS interacts with Electronic Health Record (EHR) Module through some interfaces in order to get and set data in patient's EHRs.

The Electronic Patient Record (EPR) is the record of the *periodic* care provided mainly by one institution. Typically this will relate to the healthcare provided by an hospital to a patient.

The EHR is a *longitudinal* electronic record of patient health information generated by one or more encounters in any care delivery setting. Included in this information are patient demographics, progress notes, problems, medications, vital signs, past medical history, immunizations, laboratory data and radiology reports.

The CDSS, EHR and EPR modules share a vocabulary in order to use a single terminology. This is a recurring problem for the semantic interoperability. The main used vocabularies are SNOMED [26] and ICD9 [9].

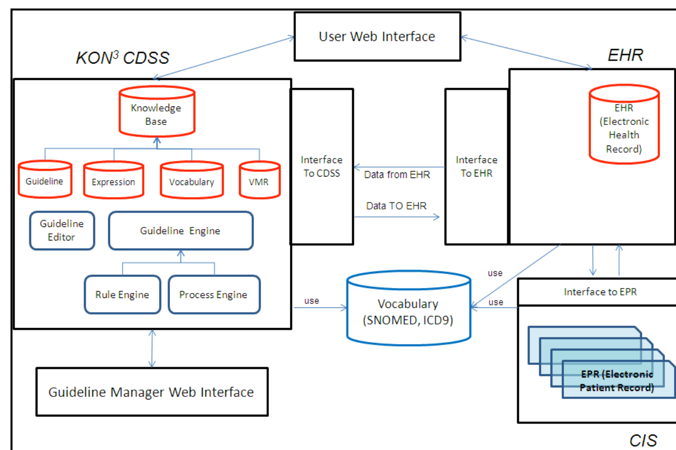

Figure 1: Kon^3 architecture

2.1. Overall picture

An overall picture describing the context on which KON^3 works is shown in Figure 2. KON^3 Guideline Engine is depicted with its knowledge base and its main modules. It is also shown a Client Module that permits clinician to interact with EHR and with KON^3 too.

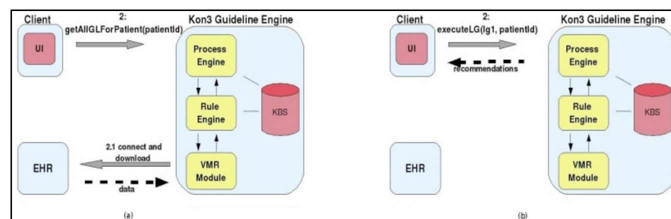

Figure 2: an overall picture

In a typical scenario, clinician inspects the electronic health record of a patient through an user interface. He also interacts with KON^3 CDSS, in order to extract guidelines that should be activated for the patient profile specified (Figure 2 (a)). During this phase, KON^3 interacts, through the VMR Module, with EHR, in order to extract needed clinical data and populate an internal VMR. After clinician and patient agree on a specific health care drift, and execute a specific guideline. So KON^3 CDSS influences health choices by clinicians in order to improve health care, offering some disease-specific recommendations (Figure 2 (b)).

3. KON^3 Knowledge Base

3.1. Guideline Model

The SAGE Guideline Model is briefly described here for illustration purposes. For a more detailed survey, see [23][25].

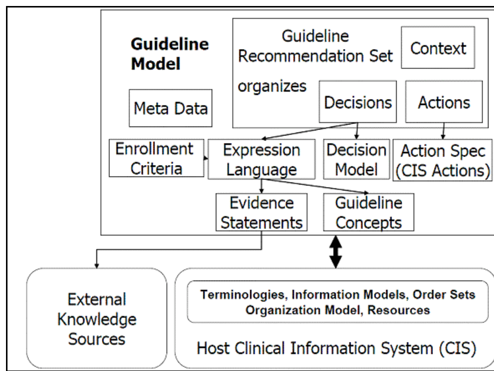

Figure 3: The SAGE Guideline Model

SAGE represents the evolution and aggregation of concepts, solutions and approaches of previous works on guideline modeling (including Asbru [15], GLIF3 [3][16], EON [23], PROforma [5], GUIDE [19] and PRODIGY [11]). It advances the state of the art by focusing on requirements that previous models have not met simultaneously: (a) incorporation of workflow awareness, (b) employment of information and terminology standards, (c) incorporation of simple flow-of-control standards, and (d) attention to integration with vendor CIS. Here we don't analyze in details the various works; for a detailed survey see [2].

The Guideline Model is a computable knowledge representation "format" for encoding the content and logic of executable CPGs.

It conceptualizes CPGs as having *metadata* such as issuing organization, *enrollment criteria* that defines its target population and *recommendation sets* consisting of some usage *context* (specific clinical circumstances) where some course of *actions* are preferred over others (*decisions* about appropriate health care).

The enrollment and decision criteria are written in terms of an executable *expression language* that make use of guideline concepts (linked to standard terminologies) and clinical evidence (formulated as *evidence statements*) for selecting particular action and that may make *queries to external knowledge sources*. Guideline actions are defined in terms of a set of *action specifications* (e.g., order laboratory tests or send an alert message) that are linked to corresponding actions in a CIS.

3.2. Standard Terminology

In order to support semantic interoperability at the domain level, vocabulary standards are needed. KON^3 uses the core vocabulary resources of SNOMED CT, in order to support semantic interoperability between CDSS and EHR.

3.3. Virtual Medical Record – VMR

A major obstacle in deploying and executing CPG is the variability of electronic medical records and the consequent need of adapters. It is also needed that a CDSS execution engine can query and update patient data during guideline execution regardless of data base organization. In order to provide standardization of content at the information model level, an idealized model of clinical record information artifacts compliant with formalisms of the HL7 Reference Information Model (RIM) [8] is adopted. This is called the Virtual medical Record (VMR) [10]. The adopted VMR in KON^3 is composed by 13 classes: *Observation, Encounter, Problem, Adverse Reaction, VMR Order, Agent, Referral, Appointment, Alert, Composite Clinical Model, Procedure, Goal.*

3.4. Expression Model

SAGE Guideline Model adopts GELLO [22] as expression language; it was developed by the Clinical Decision Support Technical Committee (CDSTC) of HL7. GELLO is a generic expression language that can be used with any object-oriented data model, but is a complex string-based language that is not easy to write for someone who is not trained technically.

To make easy for guideline encoders to author computable expressions, the adopted model introduces a number of classes that organize expressions into typed data values, variables, functions, and criteria. Each expression class corresponds to a template of stereotypical GELLO expressions.

Expression Model supports four major types of criteria templates: boolean combination, comparison, existence and goal-satisfaction. In details criteria classes supported are:

- *Comparison Criterion*, a criterion used to compare an instance or a set of instances of VMR classes - retrieved according to a coded concept and a valid window - with a predefined value;
- *N-ary Criterion,* a Boolean combination of other criteria;
- *Presence Criterion*, a criterion that checks for presence or absence of coded concept in instances of a VMR class within the valid window. There are five flavors of *Presence Criterion*, one for each specific type of instances (e.g. *Observation*), defining different behaviors: *Allergy Presence Criterion*, *Intervention Presence Criterion, Substance Administration Presence Criterion, Observation Presence Criterion*.
- *Goal Criterion*, a criterion that checks if a goal of a measurable clinical data is satisfied.

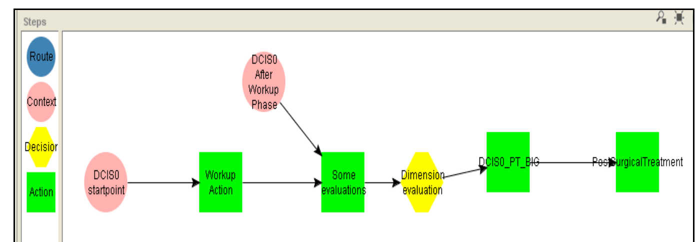

Figure 4: A guideline example design through a graph

4. KON^3 Guideline Engine

4.1. The Process Engine

The Process Engine is a module whose purpose is to execute, step by step, a guideline working as an action scheduler.

A guideline may be designed through a graph. An example of a guideline is shown in Figure 4; there are a set of useful elements to design a guideline, such as:

- **Context**, it defines the guideline context. It's possible, through the context, to define constraints, events and pre-conditions necessary to continue a guideline. It's also possible to entry in a subguideline;
- **Action**, it defines the current step to execute. The current step may be a set of actions (*Action Specifications).* Also the action may be formed by constraints and pre-conditions;
- **Decision**, it consists of an alternative sets. Each alternative is formed by rule sets. The Rule Engine evaluates these rules and it returns a Boolean value. The rules may be (A) *Strict Rule Out*, if the Rule Engine returns a *true* value, then it means this alternative isn't feasible, else it's; (B) *Strict Rule In*, if the rule sets with *true* value exceed a fixed threshold, then the alternative is strongly recommended; (C) *Rule Out*, if the Rule Engine return a *true* value, then it means that there's a contraindication, however the clinician can select the alternative; (D) *Rule In*, if the Rule Engine returns a *true* value, then the alternative is recommended. However, if there's a *false* value, the clinician can select the alternative.

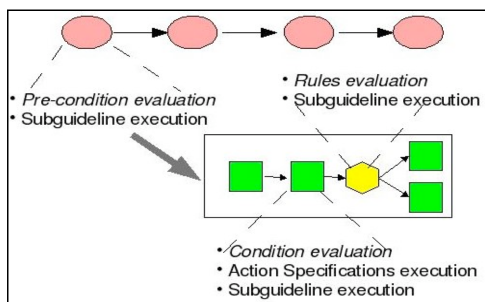

Figure 5: A typical guideline structure

In general, an expected guideline is composed by a sequence of Contexts. We adopt this convention in order to manage correctly execution state to allow at clinician to entry, in any point of the guideline. This necessity is derived by real world, e.g. a patient should start a care in a hospital and should continue it in another. So, we model these other entry points as *Contexts*, too.

Process Engine traverses a guideline graph and manages its single step. Process Engine executes each type of steps with different behavioral, see Figure 5.

If it's a *Context*, then the Process Engine controls if there are conditions to evaluate. In this case it calls the Rule Engine for evaluation of them. If the Rule Engine returns a *true* value, then this step is activated, and so the Process Engine entries in a possible subguideline. Also a subguideline is composed by other actions and decisions.

If it's an *Action*, then the Process Engine controls if there are conditions to evaluate and, if required, calls Rule Engine. If the Rule Engine returns a *true* value, then this step is activated, and so the Process Engine manages the *Action Specifications. Action Specifications* are action sets, in particular they could be classified in: (a) *Notify Action,* which purpose is to send a textual message to the clinician; (b) *Display Action,* which purpose is to display expressions, clinical data sets and supplemental materials; (c) *Inquire Action,* which purpose is to interact with the clinician to get some information about the patient; (d) *Recommended VMR Order,* which purpose is to set interventions, referrals and exams into the CIS.

If it's a *Decision*, then the Process Engine controls if there are conditions to evaluate and, if required, calls the Rule Engine. The Process Engine evaluates first the *Strict Rule Out*. If at least one is *true*, then the alternative is discarded, else the other rules are evaluated. It also determines recommended alternatives, according to *Strict Rule In* threshold and highlights them. Finally all possible alternatives are returned and proposed to clinician.

4.2. The Rule Engine

Since today there isn't a mature, complete and open source GELLO expression engine accessible via programmable interface, we choose to design and develop an *ad-hoc* engine.

The Rule Engine developed works on the expression classes defined in the Expression Model, in particular on the evaluation of the expression templates, called criteria. It is possible to define a criterion as an access precondition on *Context, Action* and *Action Specification* nodes and also as a decision rule on *Decision* nodes, such as *Strict Rule In, Strict Rule Out, Rule In, Rule Out*. Rule Engine supports and executes all four major types of criteria templates. For each class supported, there is an handler module that manages and resolves instances of it. As mentioned above, an interaction between Rule Engine and VMR module (and from VMR module and EHR, too) is strongly needed, in order to retrieve VMR instances needed to evaluate a criterion.

In a typical scenario, Guideline Engine asks to Rule Engine to resolve a particular criterion on a specified patient, e.g. a condition for evaluating if a particular *Context* may be activated. Rule Engine retrieves needed clinical data for specified patient from the VMR and resolves it. The output could be *true* (condition satisfied for the patient) or *false* (condition unsatisfied for the patient).

4.3. VMR Module

This module manages patients data collecting them into the VMRs. When asked by Rule Engine, it retrieves data for a specific patient from EHR, then it extracts only needed data and populates VMR. Finally, it offers the VMRs to Rule Engine.

5. A Case of Study: DCIS (Ductal Carcinoma In Situ)

The case of study described here, is extracted and modeled from a real guideline from NCCN[1] Clinical Practice Guidelines in Oncology v.2.2007. The specific guideline is the Ductal Carcinoma In Situ (DCIS) one.

The National Comprehensive Cancer Network (NCCN), a not-for-profit alliance of 21 of the world's leading cancer centers, is dedicated to improving the quality and effectiveness of care provided to patients with cancer. NCCN Member

[1] NCCN web site: www.nccn.org

Institutions develop resources, such as guidelines, that present valuable information to the numerous stakeholders in the health care delivery system.

DCIS refers to the most common type of noninvasive breast cancer in women. *In situ*, or "in place", describes a cancer that has not moved out of the area of the body where it originally developed. With DCIS, the cancer cells are confined to milk ducts in the breast and have not spread into the fatty breast tissue or to any other part of the body (such as the lymph nodes).

The clinical scenarios and guideline logic are encoded into a computer interpretable model of guidelines, using a Protégé [18] plug-in, called Graph Widget.

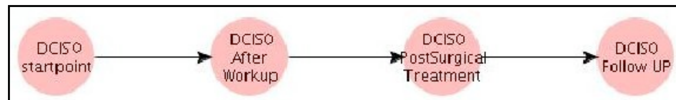

Figure 6: The DCIS guideline encoded.

The DCIS guideline, as shown in Figure 6, is composed by four main phases:

- WorkUp ("DCIS startpoint");
- Primary Treatment ("After Workup");
- PostSurgical Treatment;
- Follow Up.

WorkUp

The guideline startpoint, the first possible *Context*, is "DCIS startpoint". The specific precondition is "Female and DCIS", it is composed by two criteria: (1) A "DCIS diagnosticated" for this patient, (2) and "Patient is Female". So the precondition is *true* only if either are *true*. If this *Context* is enabled the execution could start with the subguideline. In "DCIS startpoint" subguideline there is only an *Action*, called "Workup Actions", that involves notifies that an historical and physical examination, titled "H&P to do", and a "Pathology Review" are necessary. Also two recommended clinical order titled "Mammography" and "Determination of Tumor Estrogen" should be performed.

Primary Treatment

The second phase is Primary Treatment (PT), it is composed by the subguideline shown in Figure 7. The precondition for access here is composed by the precondition for "Workup Action" *Context* and the criterion "Workup Action Completed".

Last one criterion is in order to analyze the VMR and verify that the needed clinical exams are performed in a predefined valid window.

In PT subguideline there is an inquire step ("Some evaluations" *Action*), for asking to clinician about DCIS details, such as DCIS grade and DCIS unicentrism. After a *Decision* step is performed, in order to select the recommended Primary Treatment for the patient. Here clinician responses and other VMR data (e.g. DCIS dimension) are evaluated. A summary of the underlying rules in "Dimension Evaluation" is shown in Table I.

After this step, clinician and patient select the preferred Primary Treatment from the list (*Action* DCIS0_PT_SMALL or DCIS0_PT_BIG). Examples of possible Primary Treatment for DCIS0_PT_BIG are "Mastectomy without lymph node dissection with/without reconstruction" and "Lumpectomy with RT".

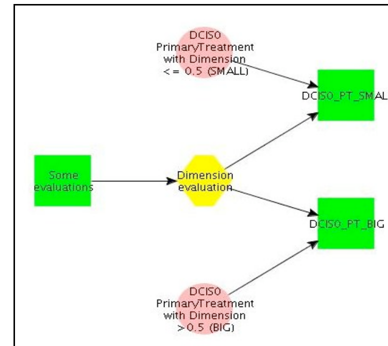

Figure 7: Primary Treatment in DCIS

	Strict Rule Out	Strict Rule In	Threshold
DCIS0_PT_SMALL	DCIS DIMENSION >= 0.5 cm	Is unicentric? Is low grade?	1
DCIS0_PT_BIG	DCIS DIMENSION < 0.5 cm	-	-

Table 1: Alternatives in Primary Treatment

PostSurgical Treatment

This phase involves the post surgical treatment, see Figure 8. In particular is determined if a specific PostSurgical Treatment is needed ("DCIS PostSurgical Treatment" *Action*), or if only the risk reduction therapy is needed, see Table II. This is determinate evaluating previous chooses, inferred from VMR (e.g. "Mastectomy" preferred as choice in Primary Treatment phase and detected as performed in VMR). Either actions are composed by some *Actions Specifications* that involves notifies to clinician, displays data, displays supplemental materials (such as other guidelines, link to publications, evidences, statistics), etc.

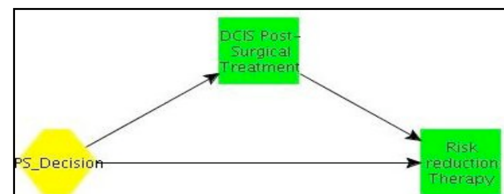

Figure 8: Subguideline in PostSurgical Treatment *Context.*

	Strict Rule Out	Strict Rule In	Threshold
DCIS0 PostSurgical Treatment	-	PT: Lumpectomy and RT PT: Mastectomy	1
DCIS0 Reduction Therapy	-	-	-

Table 2: Alternatives in Postsurgical Treatment

Follow Up

The prerequisite to access into this *Context* is a criterion composed by the PostSurgical Treatment precondition and the termination condition of the previous phase. This termination

condition is inferred from the VMR in a manner quite similar to previous ones.

In the FollowUp subguideline the required follow up exams are shown to clinicians specifying administration time interval, amount and frequency. The exams are determinate based on previous chooses, such as choices made in Primary Treatment and PostSurgical Treatment (e.g. if treated with Tamoxifen during PostSurgical Treatment, then display also, as supplemental material, "NCCN Breast Cancer Risk Reduction Guideline").

6. Conclusions and Future Works

In this paper KON^3 CDSS is described, focusing on the execution of a CPGs in oncology environment (in particular in Breast Cancer environment). However, the described system could also be applied to other domains where the activities are based on a design that entails a decision logic that is structured in an algorithmic format. In general the algorithmic format should be formed by Action Step, Decisional Step, and Context Step.

We described a Guideline Engine composed by a Process Engine and a Rule Engine to satisfy above requirements. The Process Engine is the actions scheduler, while the Rule Engine evaluates the criteria contained in a guideline. These criteria are used in order to define conditions and decision rules.

Future developments will involve: (a) *CPG Process Authoring*. It will provides a Graphical User Interface (GUI) that permits to model and formalize CPGs; (b) *CPG Rule Authoring*. It will provides a GUI that permits to define rules in a CPG; (c) *Use of EBM (Evidence Based Medicine)*. KON^3 purpose is to provide a complex system to support clinicians in their decision during a process care. The support is not only through a guideline execution, but also through EBM theory. Through EBM, the clinician is supported through a set of documents, experiences, statistics.

7. Availability

All the project documents, as Guideline Model documentation, KON^3 architecture documentation and the prototype are available on our web site *http://www.koncube.org*.

References

[1] Beal T, Heard S., "An Ontology-based Model of Clinical Information", MEDINFO2007, *IOS Press*.

[2] Boxwala A., Greenes R.A., Shortliffe E.H. et al., 2000, "Toward Standardisation of Electronic Guideline Representation". *MD Computing*, vol. 17(6), p. 39-44.

[3] Boxwala AA, Peleg M, Tu SW, Ogunyemi O, Zeng Q, Wang D, et al., 2004, "GLIF3: A Representation Format for Sharable Computer-Interpretable Clinical Practice Guidelines". *J Biomed Inform.* Vol. 37(3), p. 147-161.

[4] Ceccarelli M, Donatiello A, Vitale D, 2008, "KON3: a Clinical Decision Support System, in oncology environment, based on knowledge management", 20th IEEE ICTAI 08, Vol. 2, p. 206-210.

[5] Fox J, Das SK., 2000, *Safe and Sound*, Cambridge: MIT Press.

[6] Goldstein M.K. et al., 2004, "Translating Research into Practice: Organizational Issues in Implementing Automated Decision Support for Hypertension in Three Medical Centers", *Journal of Am. Med. Informatics Assoc.*, vol. 11, p. 368-376.

[7] Gray J. A. Muir, 1997, *Evidence Based Healthcare*, W.B. Saunders Company.

[8] HL7 Reference Information Model (RIM), *www.hl7.org/Library/data-model/RIM/modelpage_mem.htm*

[9] ICD9, *http://www.who.int/classifications/icd/en/*

[10] Johnson PD, Tu SW, Musen MA, Purves I, 2001, "A Virtual Medical Record for Guideline-Based Decision Support", Proc. AMIA Symp., p. 294-298.

[11] Johnson PD, Tu SW, Booth N, Sugden B, Purves IN, 2000, "Using Scenarios in Chronic Disease Management Guidelines for Primary Care". In *Overhage,* JM editor. Proc AMIA Symp, p. 389-393.

[12] Kim J Ah, Cho I, Kim J., 2008, "CDSS (Clinical Decision Support System) Architecture in Korea". In Proc. International Conference On Convergence and Hybrid Information Technology; p. 700-703.

[13] Lobach, D.F., Hammond, W.E., 1997, Computerized Decision Support Based on a Clinical Practice Guideline Improves Compliance with Care Standards. *Am J Med*.

[14] McGlynn E.A., Asch S.M., Adams J., Keesey J., Hicks J., DeCristofar A., and Kerr E.A., 2003, "The quality of health care delivered to adults in the United States", *New England Journal of Medicine*. Vol. 348(26), p. 2635-2645.

[15] Miksch S, Shahar Y, Johnson P., 1997, "Asbru: A Task-Specific, Intention-Based, and Time-Oriented Language for Representing Skeletal Plans". In: Motta E, Harmelen, F. v., Pierret-Golbreich, C., Filby, I.,Wijngaards, N., editor., KEML-97, p. 9-1-9-20.

[16] Peleg M, Boxwala AA, Tu SW, Zeng Q, Ogunyemi O, Wang D, et al., 2004, "The InterMed Approach to Sharable Computer-Interpretable Guidelines: A Review". *J Am Med Inform Assoc*., Vol. 11(1) p. 1-10.

[17] M. Peleg, S. Tu, J. Bury, P. Ciccarese, J. Fox, R.A. Greenes, R. Hall, P.D. Johnson, N. Jones, A. Kumar, S. Miksch, S.Quaglini, A. Seyfang, E.H. Shortliffe, and M. Stefanelli, 2003, "Comparing computer-interpretable guideline models: a case-study approach," *Journal of the American Medical Informatics Association*, Vol. 10(1), p. 52-68

[18] Protégé Official Web Site, http://protege.stanford.edu/

[19] Quaglini S, Stefanelli M, Cavallini A, Micieli G, Fassino C, Mossa C., 2000, "Guideline-Based Careflow Systems". Artif. Intell. Med., Vol. 5(22) p. 5-22.

[20] Ram Prabhu; Berg David; Tu Samson; Mansfield Guy; Ye Qin; Abarbanel Robert; Beard Nick, 2004, "Executing clinical practice guidelines using the SAGE execution engine". *Studies in health technology and informatics,* 107(Pt 1) p. 251-5.

[21] Siddiqi J. et al., 2006, "Towards and Automated Diagnosis for the Treatment of Colon Cancer: Position and Progress", IEEE AICS2006, IEEE Press. http://www.match-project.com/

[22] Sordo M, Ogunyemi O, Boxwala AA, Greenes RA, Tu S., 2004, "GELLO: An Object-Oriented Query and Expression Language for Clinical Decision Support". Summary Report prepared for OpenClinical.

[23] Tu SW, Musen MA., 1999, "A Flexible Approach to Guideline Modeling". Proc AMIA Symp 1999; *Hanley & Belfus, Inc.*, p. 420-4.

[24] Tu SW, Campbell JR, Glasgow J et al., 2007, "The SAGE Guideline Model: Achievements and Overview". *J Am Med Inform Assoc. 2007* Vol. 14(5), p. 589-98.

[25] Tu SW, Glasgow J, 2006, "SAGE Guideline Model Technical Specification", SAGE Consortium.

[26] Wang AY, Sable JH, Spackman KA, 2002, "The SNOMED clinical terms development process: refinement and analysis of content", Proc AMIA Symp 2002, p. 845-849.