Ontology Driven CPG Authoring and Execution via a Semantic Web Framework

Sajjad Hussain and Syed Sibte Raza Abidi
Faculty of Computer Science, Dalhousie University, Halifax B3H 1W5, Canada
{hussain, sraza}@cs.dal.ca

Abstract

Clinical Practice Guidelines (CPG) are used by healthcare practitioners to standardize clinical practice and to provide evidence mediated health-care. Currently, there have been considerable efforts to computerize CPG so as to operationalize them within Clinical Decision Support Systems (CDSS) and to deploy them at the point of care. In our work, we take a semantic web approach - employing a domain ontology, a patient ontology, decision rules and a rule execution engine - towards the computerization and execution of CPG for CDSS. We present an ontology-driven approach for computerizing CPG and executing them based on individual patient instances. In our work we extend the Guideline Element Model (GEM) for computerizing CPG. We have (i) defined a CPG ontology based on the Document Type Definition (DTD) of GEM for ontologically representing a GEM encoded CPG; (ii) developed CPG decision logic definition tool and defined CPG rule syntax that allows practitioners to abstract and define decision logic rules based on the CPGs decision-variables inherent within the CPG; (iii) developed a forward-chaining CPG execution engine that executes the set of CPG execution logic rules using the JENA reasoning system; and (iv) implemented an automated justification tree generation module that provides the inference trace for the solution in order to assist practitioners in understanding the rationale for the proposed recommendations. In practice, given a patient instance our CDSS is able to derive CPG based clinical recommendations. We will present a working prototype of our CPG-based CDSS for the EU Radiation Protection 118 Referral Guideline for Imaging (RPG).

1. Introduction

Clinical Practice Guidelines (CPG) are systematically developed disease-specific recommendations to assist clinical decision-making in accordance with the best evidence [5, 8]. Despite the increased efforts by medical specialists to develop medical guidelines they are still under utilized at the point of care due to a variety of behavioural, operational and technical reasons. Yet, it is widely recognized that the incorporation of CPG in the clinical workflow will have an impact of clinical decision-making. Clinical Decision Support Systems (CDSS) provide an apt medium for the computerization and execution of CPG. Yet, to date, there are numerous challenges in both the (a) computerization/codification of CPG in a formal, executable format; and (b) the systematic execution of the CPG, in conjunction with patient data, to generate evidence-based recommendations for various patient care tasks.

Functionally speaking, CDSS compare a patient's medical condition with a medical knowledge base and then guide a practitioner by offering patient and disease-specific advice [4]. In order to support CPG based CDSS, various CPG modelling methodologies have been developed to convert text-based CPG into an electronic format (Computerized Clinical Practice Guideline; C-CPG) that is both understandable by users and computers. Some of the prominent attempts include GEM [12], EON, GLIF [10], GUIDE and Prodigy [11].

We believe that C-CPG can serve not only as a framework for representing CPG electronically, but the computerization of the CPG can lead to the development of evidence-based CDSS that incorporates both domain knowledge and disease-specific recommendation/actions. By design, CPG follow a decision logic that is structured in an algorithmic format intended to support clinical decision making. We argue that the decision logic in a CPG can be used to generate explicit symbolic clinical decision-support rules for discharging specific clinical recommendations. The domain knowledge of the CPG can be represented using an Domain Ontology such that the properties in a Domain Ontology and relationships among them is explicit described and serves to both complement and validate the clinical decision support rules employed by the CDSS.

To develop CPG-guided CDSS, the challenges are (a) the apt transformation of the CPG inherent decision logic into both medically salient decision rules; (b) and to ensure the validity of the transformed knowledge and to provide trust in the recommended actions; and (c) to execute

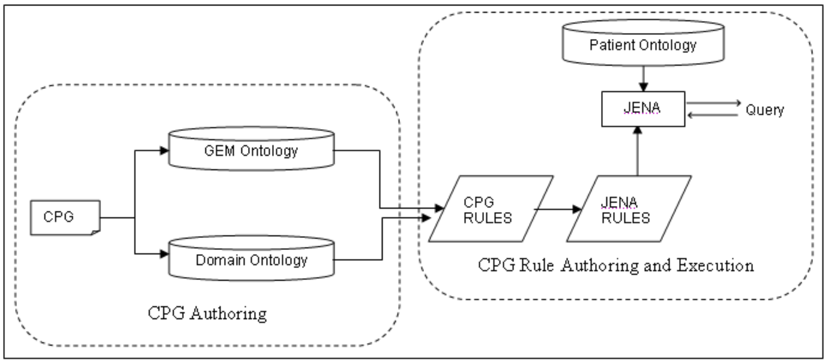

Figure 1. System Design

the computerized CPG to derive decision support. In this paper, we present an ontology-driven approach for computerizing CPG and executing them based on patient data. In our work we extend the Guideline Element Model (GEM) for computerizing CPG which currently does not support CPG execution logic definition and CPG execution. We have (i) defined a CPG ontology based GEM representation in order to ontologically represent a CPG; (ii) developed a CPG execution logic definition tool that allows practitioners to define logic rules based on the CPG decision-variables within the CPG; and (iii) developed an CPG execution engine that executes the set of CPG execution logic rules using the JENA reasoning system [6]. Given a patient instance, our CPG-based CDSS is able to operationalize the CPG and derive CPG based interventions recommendations. We will present a working prototype of our CPG-based CDSS for the EU Radiation Protection 118 Referral Guideline for Imaging (RPG).

2. Problem Description and Solution Approach

In order to establish a CPG-guided CDSS, we take the following challenges into consideration:

- How to encode CPG in a computerized format whilst encoding the underlying semantics.
- How to transform the CPGs inherent decision logic into medically salient decision rules.
- How to execute the computerized CPG to achieve decision support.
- How to ensure the validity of the transformed knowledge and to provide trust in the recommended actions.

We take a semantic web approach to meet the above challenges. The Semantic Web [13, 14] is a logic-based architecture which provides a framework for both representing and operationalizing different data sources, where data is enriched by their semantics and ontologies. For these enrichments, the Semantic Web supports standards/languages, which define data objects and relations between them using metadata vocabularies and concept hierarchies. Resource Description Framework (RDF) [13, 14] is used to annotate data objects (resources) in terms of their properties and property values as RDF triples, in an RDF graph. A property in an RDF graph is defined by its Domain (rdfs:domain) and Range (rdfs:range). The Web Ontology Language (OWL) [13, 14] is a language used for defining and instantiating ontologies in a Semantic Web. An OWL ontology includes descriptions and relationships among RDF classes, properties and their instances. We believe that by semantically annotating text-based CPGs into RDF and defining their concept hierarchies and properties in OWL, a better cooperation between CPGs in the Semantic Web can be achieved.

To develop a semantic web based CDSS, using a computerized CPG, we developed three ontologies. These ontologies model the entire working environment and knowledge used in deriving the CPG-based recommendations. The three ontologies are: (i) A CPG Ontology that models the computerized structure of the CPG. In this case we model the CPG using the GEM structure therefore our CPG ontology is based on GEM DTD (described in section 4) for representing the CPG semantically; (ii) A Domain Ontology that models the medical knowledge pertaining to the CPG. The Domain Ontology represents both the concepts described in a CPG and the relationships between these concepts as OWL classes and properties, respectively (see section 3); and (iii) A Patient Ontology that models the patient in terms of various health information elements that may constitute the longitudinal medical record of patient. The patient

Table 1. EU Radiation Protection 118 Referral Guideline for Imaging (Table A excerption)

CLINICAL PROBLEM	INVESTIGATION {DOSE}	RECOMMENDATION {GRADE}	COMMENT
Headache: chronic	XR skull, sinus, C spine (I)	Not indicated routinely (B)	Radiology of little use in the absence of focal signs/symptoms. See A13 below.
(for children see Section M)	CT (II) or MRI (0)	Not indicated routinely (B)	Some exceptions for specialists or if evidence of raised intracranial pressure, posterior fossa or other signs.
Pituitary and Juxta-sellar problems	MRI (0),	Specialised investigation (B)	Demonstration of microadenomas may not be helpful for management. CT if MRI not available. Urgent referral when vision deteriorating. Some centers use specific NM agents.
	SXR (I)	Not indicated routinely (C)	Patients who require investigation need MRI or CT.
Posterior fossa signs	MRI (0)	Indicated (A)	MRI much better than CT. CT images often degraded by beam hardening artifacts.
Hydrocephalus	CT (II)	Indicated (B)	CT adequate for most cases; MRI sometimes necessary and may be more appropriate in children. US first choice for infants. NM used in some centres, especially for shunt function.
(for children see Section M)	XR (0)	Indicated (C)	XR can demonstrate whole valve system.

ontology allows to generate standardized descriptions of a patient, which in turn serve as patient instances used to execute the decision logic. CPG decision logic is captured and represented as JENA rules (see section 5.2) that constitute elements from the Domain Ontology.

Our CDSS is divided into two main modules, namely CPG Authoring System and CPG Rule Authoring and Execution System. The CPG Authoring System requires a text-based CPG and a domain ontology eliciting the domain concepts pertinent to the CPG in question. The properties in the Domain Ontology are used to semantically annotate the decisions variables in the CPG Ontology (described in section 4.2). We encode the text-based CPG into the CPG ontology and annotate the decision variables and logic structures in the CPG Ontology based on the Domain Ontology.

CPG Rule Authoring and Execution System provides a framework for defining the decision logic rules in a CPG and executing them based on the patient clinical data. We have developed a simple rule syntax that allows practitioners to define decision rules based on the CPG decision variables (see section 5.1). We subsequently transform the decision rules into JENA rule syntax, which can then be inputted to an inference system JENA (see section 5.2 and 5.3). JENA uses the rule set to infer recommendations based on patients clinical situation. The patients clinical situation is represented in terms of a patient ontology that incorporates patient properties such as age, gender, medical history; and the values to these properties serve as input to the reasoning system. The architecture also supports the generation of an automated derivation trace of inferred recommendations for enhancing the plausibility of the judgement (see section 5.4). Figure 1 shows the architecture of our CPG guided CDSS. .

3. Domain Ontology: EU Radiation Protection 118 Referral Guideline for Imaging (RPG)

The Domain Ontology models the medical knowledge pertaining to the CPG. It represents both the concepts described in a CPG and the relationships between these concepts as OWL classes and properties, respectively. It stores the clinical scenario and treatment as instances of the Ontology. We argue that the Domain Ontology must be valid and complete in order to generate correct recommendations to the patient. To demonstrate the working of our CPG-guided CDSS, we have used the EU Radiation Protection 118 Referral Guideline for Imaging (RPG) [2] as the Domain Ontology for explaining our CPG Authoring and Execution System [3]. RPG Domain Ontology will be referred later on in the paper. Table 1 shows an excerpt on the RPG, whereas an exemplar fragment of RPG Domain Ontology is shown in Figure 2[1].

Given a clinical problem (e.g. Hydrocephalus), various investigation methods and their associated radiation dosages are proposed. For each investigation method,

[1] Taken from [2]

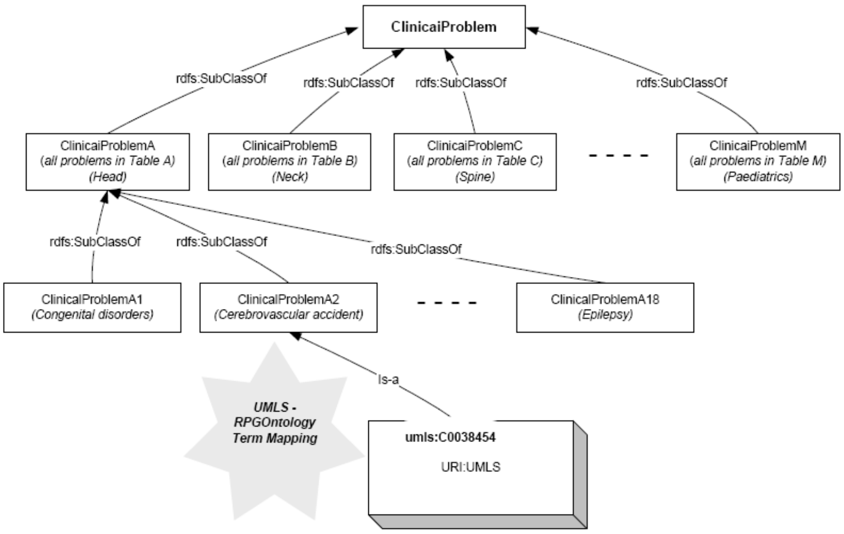

Figure 2. A fragment of RPG Domain Ontology

a recommendation is made along with the grade indicating the type of available evidence on which the recommendation is based upon. So, the correlation between the first three columns seem straightforward. The *comment* column contains discussions, cautions and alternatives of the investigation methods. Reflecting such knowledge in the ontology is the most challenging task. As shown in Table 1, a *comment* can simply state the effectiveness of investigation method (i.e. ”Radiography of little use in the absence of focal signs/symptoms”) or can suggest alternatives with implication of probabilistic reasoning.

The RPG Domain Ontology is mainly divided into three categories of concepts: *Clinical Problems*, *Investigations* and *Recommendations*. All distinct clinical problems are represented as instances of respective clinical problem classes. These classes are arranged as sub-classes under the class Clinical Problem. Investigations are radiological procedures in RPG and represented in RPG Domain Ontology similar to the Clinical Problems. For a given clinical problem, one or more investigations are recommended. Recommendations are treatments based on investigations and clinical problems. Each recommendation is represented along with the grade of its evidence. An *indicated* recommendation is most likely to contribute to the diagnosis and management, while a *not indicated routinely* recommendation emphasizes the limitations of an investigation.

4. CPG Authoring System

In order to support CDSS, various CPG computerization methodologies have been developed to convert text-based CPG into an electronic format (for instance GLIF, GEM, GASTON), most CPG methodologies do not extend as far as the execution of the computerized CPG based on a patient’s case; this is typically due to the absence of a CPG execution engine. In our work we aim to execute the CPG, and in this regard the challenges that we addressed are: (a) the apt transformation of the CPGs inherent decision logic into both medically salient decision rules; (b) to ensure the validity of the transformed knowledge and to provide trust in the recommended actions; and (c) to execute the computerized CPG to derive clinical decision support. In order to meet above mentioned challenges, we design a CPG Authoring framework that incorporates a CPG ontology, based on the GEM DTD, to represent the CPG in concert with the domain ontology. The CPG authoring tool allows the user to computerize a text-based CPG by annotating the decision variables inherent within the CPG based on the concepts described in the domain ontology.

4.1 Guideline Elements Model (GEM)

The Guideline Elements Model (GEM) provides a platform for annotating text-based clinical practice guidelines (CPG) in Extensible Markup Language (XML) documents and representing main/key features in those CPG [12]. For this purpose, GEM incorporates over 100 elements in order to fully implementation heterogeneous parts that make up the content of a clinical practice guideline. These elements are organized in the GEM hierarchy. The Knowledge Component elements are the most important elements of GEM and represent procedural, conditional or imperative knowledge found in a CPG. The

Knowledge Component element contains Recommendation sub elements, each of which describes the recommended actions for patients. These recommendations can be either imperative or conditional. Imperative recommendations are those that are applicable to the entire eligible population. In contrast, conditional recommendations describe decision variables that need to be considered and actions to be undertaken if the decision variables meet a certain criteria. An important sub-element of conditional recommendations is the logic element. The logic element states explicitly the conditions that are required for certain actions to take place.

4.2 CPG Ontology

We developed the CPG Ontology based on the GEM DTD using an Ontology Editor Protégé [9]. We added a new property *variable.name* in the *decision.variable* class into the CPG Ontology, where the *rdfs:domain* of variable.name is the decision.variable class in CPG Ontology and *rdfs:range* is all the properties in the pre-defined Domain Ontology. By annotating each decision.variable with a property variable.name with the property value from the Domain Ontology, each decision variable in the GEM encoded CPG is represented by a semantically annotated variable.name and related with other resources and properties in the Domain Ontology. Figure 3 demonstrates the semantic annotation of decision variables `dv1 ...dv11` and action variable `a1` with the list of variable names, which are represented as properties in the RPG Domain Ontology.

In Figure 3, each of the decision variables in the CPG Ontology are annotated with the properties in the RPG Domain Ontology based on concepts presented in Table 1. For example, decision varible `dv1` is annotated with a property *hasClinicalProblem*, where the property value describes various clinical problems in the RPG test-case as shown in the first column of Table 1. Similarly, decision variables `dv2, dv4, dv5, dv10, dv11` are annotated with properties, where the property values describe one (represented by *applyOnlyTo*) or more investigations based clinical problems as shown in the second column of Table 1. Finally, `dv8, dv9` are annotated and describe the recommendations along with their grades based on investigations and clinical problems as shown in the third column of Table 1. Action variable `a1` is annotated with a property `isRecommended`, where the property value repesents the list of investigations along their methods and recommendations along their grades.

4.3 Patient Ontology

Patient Ontology models the Electronic Patient Record (EPR) architecture patient profiles in terms of various health

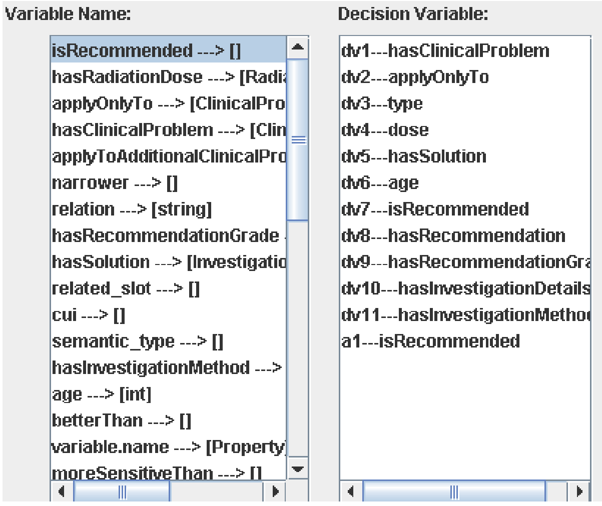

Figure 3. Annotation of Decision Variables and Actions based on RPG Domain Ontology

information and patient clinical situations. It allows to generate standardized descriptions of a patient, which in turn serve as patient instances used to execute the decision logic of a CPG within the CDSS.

5. CPG Rule Authoring and Execution System

The functionality of the CPG Rule Authoring and Execution system is to encapsulate the clinical decision making logic inherent within a CPG in terms of logical rules that can be executed by reasoning engines, to derive CPG-based recommendations for specific patient conditions. To achieve this functionality we built two sub-modules namely, Rule Authoring Module and Execution Module for defining decision logic rules embedded in a CPG, and executing them based on clinical investigations and patient profiles, respectively.

The Rule Authoring Module provides an interface (see section 6) to the practitioners to define decision logic rules a CPG Rule Syntax. This is achieved as follows:

- A CPG rule is written in the logic tag of CPG Ontology and comprise decision variables present in the CPG ontology.
- Each of the decision variables has a property *variable.name*, where the property value (of each *variable.name*) corresponds to a property in the Domain Ontology (see Example 4.1).

Upon completion of the rule authoring process, the rule authoring module transforms the CPG rules into the JENA

```
Logic := IF Decision_Variable_List THEN Action_Variable
       | Action_Variable
Decision_Variable_List := dv Rel Node, dv Rel Node, .., dv Rel Node
                                            //where <dv,dv_name> ∈ V
Action_Variable := a Rel Node               //where <a,a_name> ∈ A
Rel := < | <= | > | >= | =
Node := ?                  // variable
      | 'a literal'        // a plain string literal
      | Algebra
      | [dv1 dv2 ...  dvn]
Algebra := Value | Value + Value | Value  Value | Value * Value
Value := dv | number        // dv must already been declared before
```

Figure 4. CPG Rule Syntax

syntax for their execution with the JENA inference engine (see section 5.3).

The Execution Module invokes the JENA inference engine to execute a CPG, and infer recommendations based on patient clinical situations, various treatment plans modelled in CPG and its Domain Ontology. We model instances from the Domain Ontology, CPG Ontology and Patient Ontology as RDF graphs, which serve as the knowledge base for JENA. JENA inference engine starts with the knowledge base and the JENA rule set and builds an inference model. The model is then used for querying inferred recommendations using backward logic programming engine (described in section 5.4). Furthermore, we use the JENA inference model and their supported modules for presenting the derivation trace for inferred recommendations. The detailed process steps are described in following subsections.

5.1 CPG Rule Syntax

Let,

V is the set of pairs `<dv,dv_name>` of all decision variables and their names, where `dv` is a decision.variable instance in the CPG Ontology and `dv_name` is the variable.name property value of `dv`,

A is the set of pairs `<a,a_name>` of all action variables and their names, where `a` is an action.variable instance in the CPG Ontology and `a_name` is the variable.name property value of `a`.

Rules in the logic element of CPG Ontology can be written in the CPG rule syntax as shown in Figure 4. Each rule is a forward rule, which has a list of decision variables (body) and an action variable (head) of the rule, followed by `IF` and `THEN`, respectively. In the decision variable list, each variable `dv` is an equality or inequality relation with either i) a variable, ii) a string, iii) a list of (already declared) decision variables or iv) an algebraic (binary) formula. An example CPG rule is described in Example 4.1.

5.2 JENA Rule Syntax

JENA is a general purpose rule-based reasoner used to implement both the RDF and OWL reasoners and also can be applied to general purposes [6]. This reasoner supports rule-based inference over RDF graphs and provides forward chaining, backward chaining and a hybrid execution model. JENA is comprised of two internal rule engines, namely, forward chaining RETE engine and backward logic programming engine and can run as a backward chaining reasoning system. An informal description of the simplified text rule syntax (as mentioned in JENA documentation [6]) is shown in Figure 5. The ",'' separators are optional. The *functor* in an extended triple pattern is used to create and access structured literal values. An example rule written in JENA rule syntax is shown in Example 4.1.

5.3 Transformation of CPG rules into JENA rules

Let,

X is the set of JENA `?varname`

$\prod = V \cup A$

$D : \prod \mapsto X$ is a function which takes either a decision variable or action variable and returns a unique JENA variable, which represents resource of its variable name.

$R : \prod \mapsto X$ is a function which takes either a decision variable or action variable and returns a unique JENA variable, which represents value of its variable name.

$\nu : \prod \mapsto X$ is a function which takes either a decision variable or action variable and returns the encoded value.

Each CPG rule have a `Decision_Variable_List` and an `Action_Variable`, which are followed by the `IF` and `THEN` and serve as head and body of the CPG rule, respectively. Translation of CPG rules into JENA rules is performed by a *Transformation Algorithm*. It parses the head and body of a CPG rule and translate the decision variable relations (in the body) and action variable (in the

```
Rule := bare-rule | [ bare-rule ] | [ ruleName :  bare-rule ]
bare-rule := term, ...  term -> hterm, ...  hterm        // forward rule
          | term, ...  term <- term, ...  term        // backward rule
hterm := term | [ bare-rule ]
term := (node, node, node)        // triple pattern
      | (node, node, functor)        // extended triple pattern
      | builtin(node, ...  node)        // invoke procedural primitive
functor := functorName(node, ...  node)       // structured literal
node := uri-ref        // e.g.  http://foo.com/eg
      | prefix:localname        // e.g.  rdf:type
      | ?varname           // variable
      | 'a literal'        // a plain string literal
      | 'lex'typeURI        // a typed literal, xsd:* type names supported
      | number        // e.g.  42 or 25.5
```

Figure 5. JENA Rule Syntax

***Transform** (R)*

Let R be a CPG rule, B be a body (premises) and H be a head (conclusion) of JENA rule.

1. Parse the body of a CPG rule R.
   (a) For each decision variable relation in the body of R.
      i. If the decision variable dv has a relation with its value v and annotated with a variable name `dv_name` then add the triple $(D(dv)$ `dv_name` $v)$ in B.
      ii. If the decision variable dv has a relation with another (declared) decision variable dv' and annotated with a variable name `dv_name` then add the triple $(D(dv)$ `dv_name` $\nu(dv'))$ in B.
      iii. If the decision variable dv has a relation with a variable `?` and annotated with a variable name `dv_name` then add the triple $(D(dv)$ `dv_name` $R(dv))$ in B.
      iv. If the decision variable dv has a relation with a variable list of variable $[dv_1 \ldots dv_n]$ and annotated with a variable name `dv_name` then add the triple $(D(dv)$ `dv_name` $List(\nu(dv_1) \ldots \nu(dv_n)))$ in B.
   (b) For decision variables with inequality relation and algebraic formula, repeat steps (i - iv) and add JENA built-in functors for inequality and algebraic formula in B.
2. Parse the head of a CPG rule R : analogous to step (1).

Figure 6. CPG Rule Transformation Algorithm

head) into JENA rule syntax, recursively. Main steps of this algorithm are outlined in Figure 6 and illustration is presented in Example 4.1.

Example 4.1

Let R be a decision rule written for the RPG test-case in CPG rule syntax.

```
R = IF dv3=Patient, dv1=?, dv6>45, dv5=?,
dv8=indicated, dv9=?, dv10=?, dv11=? THEN
dv7=[dv5 dv11 dv8 dv9]
```

The above CPG rule is translated into JENA rule and shown as follows:

```
Transform (R) = [conditional1:
(?X1 rdf:type rpg:Patient) , (?X1
rpg:hasClinicalProblem ?X3) ,
greaterThan(?X6, 45) , (?X1 rpg:age
?X6) , (?X3 rpg:hasSolution ?X4), (?X4
rpg:hasRecommendation rpg:indicated) ,
(?X4 rpg:hasRecommendationGrade ?X7) ,
(?X4 rpg:hasInvestigationDetails ?X8)
, (?X8 rpg:hasInvestigationMethod ?X9)
(?X1 rpg:isRecommended List(?X4 ?X9
rpg:indicated ?X7))]
```

In Example 4.1, `dv1=?` is a decision variable relation, where `dv1` is annotated by a variable name `hasClinicalProblem` (as shown in Figure 3) and has equality relation with a variable `?`. The decision variable relation `dv1=?` is translated into JENA syntax as `(?X1 rpg:hasClinicalProblem ?X3)` and added in the body of the JENA rule via step (iii) of the transformation algorithm (as shown in 6). Similarly,

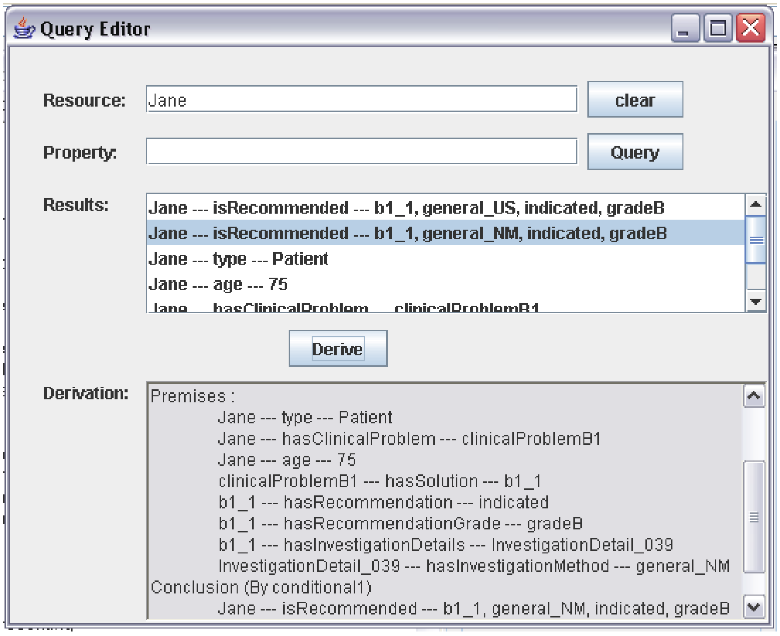

Figure 7. Derivation Trace for Recommendations for Jane

the decision variable relation `dv6>45` is translated (via step i) into `greaterThan(?X6, 45) , (?X1 rpg:age ?X6)`. Note that JENA built-in functor `greaterThan(?X6, 45)` is added (via step (b)) due to inequality relation.

5.4 Automated Derivation Trace Module

We developed an automated derivation/justification module which generates the justifications behind inferred recommendations based on the CPG and the patient data. This is to provide a trace of the rule execution to the medical practitioner so that he/she may be able to interpret the logic behind a certain recommendation; without such justifications the system will turn into a 'black-box' which is not appreciated by medical practitioners. The derivation includes the linear representation of premises (facts) under which the JENA rules are satisfied and the conclusions based on those rules. This module takes a derived patient recommendation (derived facts) from the JENA model (knowledge base) and generates facts which served as premises for deriving the patient recommendation, recursively. This module terminates, if all the premises are ground instances (known facts).

In the RPG test-case, based on the encoded JENA rule in Example 4.1, automated derivation trace for the recommendations for a patient "Jane" is generated and shown in Figure 7.

6. CPG Authoring and Execution System (CPG-EX)

We used the above presented approaches to establish an Ontology-Driven CPG Authoring and Execution System (CPG-EX). The System inputs a text-based CPG and loads its pre-defined Domain Ontology and the Patient Ontology. It encodes the CPG in terms of the CPG Ontology. It uses the Domain Ontology to semantically annotate the decision variables in the CPG Ontology (as described in section 4.2). Subsequently, it transforms the CPG rules encoded in CPG Ontology into JENA rule syntax and passes the rules to the JENA reasoning system. Finally, the CPG-EX system invokes JENA reasoner with the patient instance and rule set for inferring recommendations and other information based on the patient profiles. The system also generates derivations traces for inferred recommendations in order to enhance plausibility of those recommendations.

The CPG-EX interface (as shown in Figure 8) is composed of three panels. On the left panel, user can load a CPG. The middle panel has the CPG Ontology structure for semantically annotating the CPG into CPG Ontology. The CPG Ontology panel further provides the following features to the user:

- *Duplicate Button*: Duplicates selected tag/element in the CPG Ontology structure.
- *Save Button*: Save the CPG Ontology structure into a file in RDF/XML format.

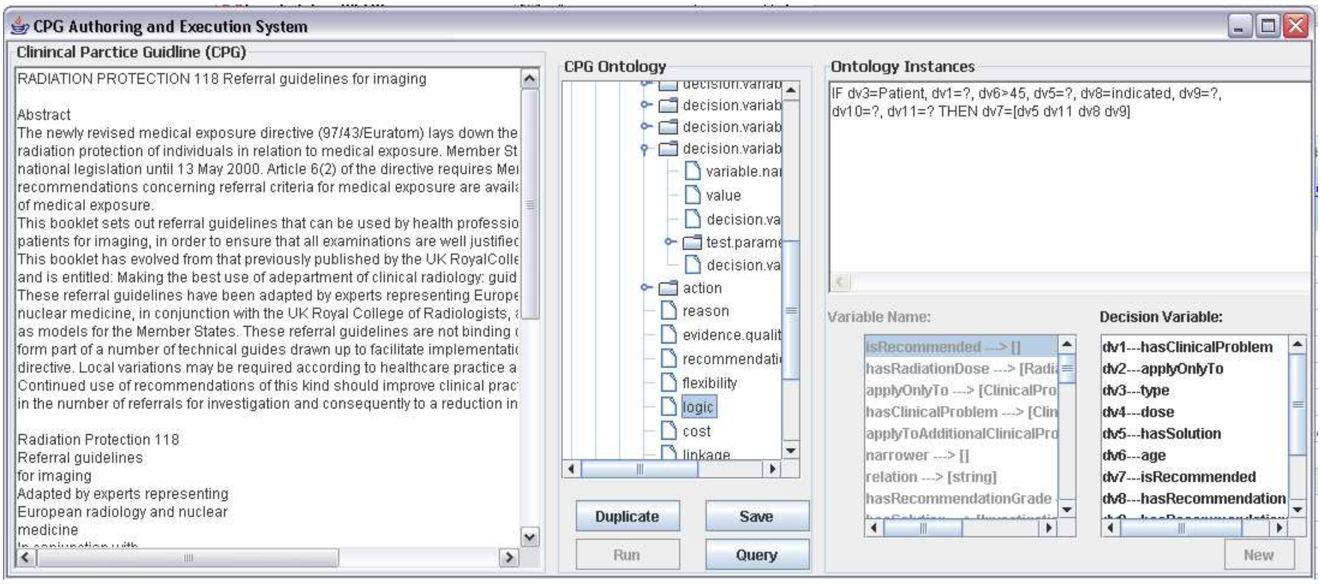

Figure 8. CPG-EX Interface

- *Run Button*: Run the CPG on patient instances by transforming CPG logic tags into JENA rules.
- *Query Button*: Query the result after running CPG.

Right panel is used for assigning instances to each tag in the CPG Ontology structure. It consists of *Ontology Instances* text box, *Variable Name List* and a *Decision Variable List*. The Variable Name List displays all the properties stored in the Domain Ontology. The Variable Name List becomes active only when a variable.name tag (from CPG ontology) is selected. User can select a variable name form the Variable Name List and assign it to a decision variable. Decision Variable List shows the list of annotated decision variables associated with their variable names. Query Button initiates a Query Window (as shown in Figure 7). Query Window provides a user to query patient information and inferred recommendations. Furthermore, it allows the user to view the derivation trace for inferred recommendations.

CPG Authoring is done by extracting textual information of the text-based CPG (shown in the left panel of CPG-EX) and annotating them based on the classes and their properties in the CPG Ontology (shown in the middle panel of CPG-EX). The annotated text is assigned to the Ontology Instance text box (shown in the right panel of CPG-EX) based on the classes/properties in the CPG Ontology. Rule Authoring is performed by defining decision rules in the logic tag of CPG ontology. Decision rules are written in CPG syntax. Rule Authoring process can be outlined by following main steps:

1. Select the list of decision variables from the Decision Variable List, which represents the body (premises) of the rule and followed by `IF`.
2. Select the action variable from the Decision Variable List, which represents the head (conclusion) of the rule and followed by `THEN`.
3. For each decision variable and action variable in the rule, an equality/inequality relation can be defined with either a variable, a value, a binary algebraic formula, another decision variable or list of decision variables (see Example 4.1).

We tested our system on the clinical practice Guideline, EU Radiation Protection 118 Referral Guideline for Imaging (RPG) [3]. We used the pre-defined RPG Ontology (see section 3) and Patient Ontology, which are defined and described in [2]. The CPG-EX system encoded the text-based RPG into the CPG Ontology and transformed the CPG rules into JENA. Finally, it invokes the JENA reasoner to query inferred recommendations and other information based on selected patient profiles. An example recommendation for the patient *Jane* based on RPG test-case is shown in Figure 7.

7. Discussion and Concluding Remarks

In this paper we developed a CDSS by exploiting C-CPG and Domain Knowledge represented as ontologies. The proposed attempt could also be applied to other similar domains (such as workflows) where the activities are based on a design that entails a decision logic that is structured in an algorithmic format.

In addition to the logic tag, CPG Ontology is furthermore enriched by the *Algorithm* tag, which represents sequential stages in health management described by a CPG. The Algorithm tag is comprised of *Action Step*, *Conditional Step*, *Branch Step* and *Synchronization Step*. We believe that exploiting the Algorithm tag is beneficial for improving the CPG-EX and formally representing the sequential stages of recommendations and then executing them based of patient clinical situations and recommendation stages.

CPG rule authoring module provides an interface for defining rules in CPG rule syntax. Although, this rule syntax helps practitioners to define abstract decision rules and refrains from the formal details. However, this syntax is not as expressive as formal languages such as N3 and RDF [14]. This is a common trade-off and challenge while building rule authoring interfaces. There has been an attempt to define an abstract syntax ARDEN for representing medical decision logic [7]. It would be interesting to investigate the rationale and expressiveness of ARDEN syntax and then exploit this syntax for authoring the decision rules for CPGs in CPG-EX.

So far, we have proposed an approach which takes into account one CPG and its corresponding domain ontology for establishing the CPG execution system. However in practice, practitioners may require more than one CPG to consult and advice better and feasible recommendations. Hence, it is important to investigate and establish such a framework, which allows multiple CPG with their inherent logic and decision structures to support in finding patient specific recommendations in a consistent way [1].

We believe that whilst working with a CDSS, the presentation of inferred information without providing justifications reduces the plausibility of the inferred recommendations to the practitioner. Although CPG-EX provides justifications/proof trees at their calculi levels, these proofs are not understandable and difficult to read. In order to share these proofs among users and other systems, proof generation and natural language presentation of proofs at the higher level of abstraction and granularity are necessary.

In this paper we established a CDSS and showed how the Domain Ontology of a CPG can be linked with CPG Ontology for semantically annotating the decision variables and decision logic elements in CPG Ontology. Finally, we defined a transformation of CPG Rules (annotated in the CPG ontology) into JENA Rules, which are used by JENA inference engine to infer recommendations and other information based on patient profiles. We implemented an automated derivation/justification generation module that provides the inference trace for the solution in order to assist practitioners in understanding the rationale for the proposed recommendations.

Future developments will involve: i) developing methods for incorporating multiple CPGs to ensure more feasible clinical decision making, ii) adapting the Algorithm tag for representing sequential steps for recommendations, iii) exploiting ARDEN syntax for authoring CPG decision rules and iv) presenting justifications for clinical recommendations in natural language at higher level of abstraction and granularity.

References

[1] S. Abidi. Medical knowledge morphing: Towards the integration of medical knowledge resources. *18th IEEE International Symposium on Computer-Based Medical Systems, Dublin*, June 23-24 2005.

[2] H. Chen, S. Devitt, and J. D. Roo. Rpgontology: A semantic webized knowedge base for radiation protection guidelines. *First International Conference on Semantic, Knowledge and Grid, Beijing, China*, Nov. 27-29 2005.

[3] E. Commission. Radiation protection 118 referral guideline for imaging. 2000.

[4] M. Eccles, E. McColl, N. Steen, N. Rousseau, J. Grimshaw, and D. Parkin. Effect of computerized evidence based guidelines on management of asthma and angina in adults in primary care: cluster randomized controlled trial. *BMJ*, (325):1–7, October 2002.

[5] M. J. Field and K. N. Lohr. *Clinical Practice Guidelines: Directions for a New Program*. National Academy Press, Institute of Medicine, Washington, DC, 1990.

[6] HP-Lab. Jena: Semantic web framework. *http://jena.sourceforge.net/documentation.html*.

[7] G. Hripcsak. The arden syntax for medical logic modules: introduction. *Computers in Biology and Medicine*, 5(24):329–30, 1994.

[8] R. A. Jadad and R. B. Haynes. Collaboration-advances and challenges in improving evidence-based decision making. *Med Decision Making*, (18):2–9, 1998.

[9] H. Knublauch, R. W. Fergerson, N. F. Noy, and M. A. Musen. The protégé owl plugin: An open development environment for semantic web applications. *In McIlraith et al*, pages 229–243.

[10] M. Peleg, A. Boxwala, and O. Ogunyemi. Glif3: The evolution of a guideline representation format. *AMIA Annual Symposium*, pages 645–649, 2000.

[11] M. Peleg and S. Tu. Comparing computer-interpretable guideline models: A case-study approach. *J Am Med Inform Assoc*, (10):52–68, 2003.

[12] R. N. Shiffman, B. T. Karras, A. Agrawal, R. Chen, L. Marenco, and S. Nath. Gem: A proposal for a more comprehensive guideline document model using xml. *J Am Med Informatics Assoc*, (7):488–98, 2000.

[13] W3C. Owl: Web ontology language overview. *http://www.w3.org/TR/owl-features/, W3C Recommendation*, February 2003.

[14] W3C. Rdf primer. *http://www.w3.org/TR/rdf-primer, W3C Recommendation*, February 2004.