2008-01-01

Software Agents Representing Medical Guidelines

John McGrory
Technological University Dublin, john.mcgrory@tudublin.ie

Jane Grimson
Trinity College, jane.grimson@tcd.ie

Frank Clarke
Technological University Dublin, frank.clarke@tudublin.ie

See next page for additional authors

Recommended Citation

McGrory, J., Grimson, J., Clarke, F., Gaffney, P.: Software Agents Representing Medical Guidelines HEALTHINF 2008 International Conference on Health informatics, , Funchal, Madeira, Portugal, 2008.

Funder: Technological University Dublin

Authors
John McGrory, Jane Grimson, Frank Clarke, and Peter Gaffney

SOFTWARE AGENTS REPRESENTING MEDICAL GUIDELINES

John McGrory, Frank Clarke
Dublin Institute of Technology, Dublin 8, Ireland
john.mcgrory@dit.ie, frank.clarke@dit.ie

Jane Grimson
School of Computing, Trinity College Dublin, Dublin 2, Ireland
jane.grimson@tcd.ie

Peter Gaffney
Adelaide & Meath Hospital, Incorporating the National Children's Hospital (AMNCH), Tallaght, Dublin 24, Ireland
peter.gaffney@amnch.ie

Keywords: Clinical laboratory validation, patient-centred, patient-focused, agents, computer-interpretable-guideline (CIG), guidelines and protocols. Software Agents representing medical guidelines

Abstract: Guidelines are self-contained documents which healthcare professionals reference to obtain specific disease or medical condition knowledge for a particular population cohort. They view these documents and apply known facts about their patients to access useful supportive information to aid in developing a diagnosis or manage a condition. Traditional CIG models decompose these guidelines into workflow plans, which are then called using certain motivational trigger conditions controlled by a centralised management engine. Therefore, CIG guidelines are not self-contained documents, which specialise in a particular condition or disease, but are effectively a list of workflow plans, which are called and used when the patient information is available. The software BDI agent offers an alternative approach which more closely matches the modus operandi of narrative based medical guidelines. An agent's beliefs capture information attributes, plans capture the deliberative and action attributes, and desire captures the motivational attributes of the guideline in a self-contained autonomous software module. This synergy between the narrative guideline and the BDI agent offers an improved solution for computerising medical guidelines when compared to the CIG approach.

1 INTRODUCTION

Clinical guidelines are condition focused documents through which domain specific aims, goals, procedures, plans and normal reference ranges are disseminated to healthcare professionals (Browne, 2005) (Oosterhuis, 2004). To maximise influence on a market audience these documents are written in domain specific languages and ontologies (e.g. cardiology, neurology and paediatrics). The purpose of these documents are to guide the reader, and streamline activities around a particular medical condition, organ or disease using evidence based supportive information. When a clinical or laboratory guideline is developed by an expert group they focus on best practice for the specific disease, condition or organ. They include all relevant knowledge, logic and motivational aspects they deem necessary to adequately describe the domain. Therefore, guidelines are autonomous self-contained documents, which can be used in whole or in part, to provide supporting information for healthcare decision-making, once their meaning is not taken out of context.

Clinicians and laboratory technologists care for patients so it is their responsibility to filter through these guidelines acting on a patient's behalf. They must make use of the maximum decision-making support offered by these resources based on the known facts about the individual. This is termed *patient-centred validation*. However, the enormous quantity and presentation style of these documents makes it difficult for professionals to quickly identify relevant guidelines, and extract information and intended logic contained within them, and apply them usefully in a patient-centred fashion (Peleg et al., 2003).

One approach to overcome these issues is to use Computer-Interpretable-Guidelines (CIG). The underlying operation of CIG models is to decompose the narrative guideline into separate workflow activities and management rules. The management rules from each guideline are linked together centrally using an inference engine. This inference engine is constructed by the CIG developers using rules that link the various workflow activities together. These management rules provide the motivation for a centralised inference engine to choose particular workflow activities depending on the patients known characteristics (e.g. weight, gender, height). As more guidelines are decomposed and added, the number of workflow activities and the size and complexity of the centralised inference engine increases. Using this technique the original self-contained knowledge developed by the guideline author's is fragmented and the true context is lost.

Agent-oriented software operates on the notion that human intelligence and decisions can be synthesised by managing tangible elements, such as beliefs, goals, sub-goals, plans and intentions. Each agent is an autonomous software module that is self-contained, has its own inference engine and a set of beliefs, goals and plans. The goals, beliefs and plans are encoded into the agent and the inference engine interprets them. When a goal is chosen the inference engine searches plans that match currently known beliefs and prioritises them. The most appropriate plan is selected and executed to perform some activity or choose another goal. In addition to its inference engine the agent approach also permits separate agents to socialise, work in groups and collaborate to solve common goals.

There are characteristic similarities between guidelines and a particular type of agent, namely the Belief-Desire-Intention (BDI) agent as shown in Table 1.

BDI Structure		**Characteristics**		**Guideline Structure**
Belief	↔	Informational attributes	↔	Facts
Desire	↔	Motivational attributes	↔	Goals and aims
Intension	↔	Deliberative attributes	↔	Selection of Actions

Table 1: BDI to Guideline mapping.

By encoding the agent with the knowledge, logic and motivational components of a guideline, it creates an opportunity for an individual agent to represent a single guideline. As additional guidelines are added, new agents are created to represent them. This technique means that the original self-contained knowledge developed by author's remains as an integrated software module and the context is retained.

The thrust of this paper is to demonstrate that a single software agent possesses the ability to reproduce the function and content of a clinical guideline as a 1:1 ratio. This is because a synergy exists between guideline and agent characteristics.

2 COMPARISON OF GUIDELINES, CIG'S AND AGENTS

Guidelines play a crucial supporting role in the detection, diagnosis, treatment, and supervision of diseases in patients, in a modern healthcare setting. In some cases these documents aid in the planning of treatments, quantifying of medication amounts, monitoring of patient responses, which could all have serious patient safety issues if inaccurate information was used. Therefore, it is vital the knowledge, logic and motivation contained within the guideline are clearly understood with reference to a specific patient. But how do CIG's and agents reproduce the knowledge, logic, action and more specifically motivation of guidelines? In the following subsections the function, use and methodology behind guidelines, CIGs and agents are discussed.

2.1 Guideline Infrastructure

Guidelines are autonomous literature sources which seldom rely on any other knowledge, except some base-level understanding. They provide information, logic and motivation in order to describe aspects of a particular domain. The guideline document itself is made up of a series of aims, goals, procedures and plans. It was never expected guidelines would be used exclusively in a form of cookbook medicine, but more in a complementary and supportive role by providing domain specific information. Therefore, these documents are rarely used in whole, but more in part, to aid in healthcare delivery. This means that the guideline should be viewed holistically, and not just as a collection of separate workflow elements taken out of context.

In summary, a clinical guideline is a self-contained document which specialises in a specific condition, organ or disease. Medical professionals interface with this document in light of the known patient information to support their patient-centred healthcare delivery.

2.2 Computer-Interpretable-Guidelines

The underlying principle behind the CIG approach is to disassemble the guideline into separate workflow activities, and then orchestrate rules to link these activities together centrally based on the presented patient data. The majority of CIG's decompose the medical guideline into an Arden Syntax Medical Logic Module (MLM) or similar by dividing the narrative guideline into '*evoke*', '*data*', '*logic*' and '*action*' slots. These slots are used to develop a software workflow activity plan, complete with a triggering condition, but contain no motivation or goal. As more guidelines are added, rules controlling the selection, merging and division of workflow activities increases. The centralised CIG control engine which manages these activities provides the motivation or goal for the software to operate. As all guidelines are coupled together via the centralised set of management rules it is not possible to distribute this activity among a number of computer systems. This is fundamentally different to the true operation of a medical guideline where many copies of the same guideline exist.

In summary, a CIG is a list of workflow plans which are called and used when the relevant patient information is available. It is centralised and the original knowledge is fragmented and cannot be truly interfaced with.

2.3 Software agents

An agent is an autonomous self-contained software module which is programmed using belief, goal and plan attributes. Each agent has its own inference engine which interprets these attributes in order to perform some activity. The principle of a BDI agent's operation is based on a belief capturing the informational attributes, the desire capturing motivational attributes and the intention capturing the deliberative attributes of an agent (Rao et al., 1995). An agent shell is a generalised version of an agent which can be adapted for a wide variety of different applications. There are a number of BDI agent shells available such as Jason, 3APL and Jadex. Jadex 0.932 was the agent shell used in this research. Although there are characteristic similarities between agents and guidelines the raw Jadex shell is not capable of capturing guideline functionality without some modification. To this end the Jadex agent shell was adapted and this modified version of the Jadex agent was titled Autonomous Socialising Knowledge agent (ASK-agent).

Using the ASK-agent model a guideline can be decomposed into workflow activities, but instead of having a centralised inference engine to manage all the guidelines, each guideline has its own inference engine. The guidelines have the ability to communicate with other resources using message passing. The ASK-agent encodes the separate workflow paths and motivational management rules of the guideline within a single autonomous software module. There is no centralised engine managing the separate ASK-agents, therefore the approach is distributed. Each agent registers itself, complete with the services it provides, language and ontology it uses with a Directory Facilitator (DF). The DF acts as a *goldenpages* for agents, allowing them to be looked-up, accessed and used as independent resources.

Adapted MLM Slot	**ASK-Agent Component**
Evoke	The ASK-agent's action trigger to perform some task, or perform goal.
Data	Belief, the facts the agent uses to trigger logic. A belief can be any Java object.
Logic	Condition, precondition or trigger based on beliefs used for the selection of an appropriate workflow activity plan or goal.
Action	Execution of the workflow activity.
**NEW* Achieve Goal*	Motivates the ASK-agent to achieve a specific goal to reach some desired state, such as determine patient gender, age, PatientID.
**NEW* Maintain Goal*	Motivates the ASK-agent to maintain a specific condition (e.g. maintain the plausibility value above 60%)
**NEW* Query Goals*	Motivates the ASK-agent to seek alternative avenues on an IF basis without committing to the workflow activity (e.g. test alternative paths before committing to the path, such as would knowing the gender of the patient alter the outcome?).
**NEW* Meta-level reasoning*	If more than one workflow activity is triggered simultaneously, but only one should be chosen the ASK-agent must choose the most appropriate course of action to take.
**NEW* Modal reasoning*	If data stored in the beliefs or received in a message has a level of truth, but cannot be established as 100% true or false, then the ASK-agent can weight its selection of an appropriate action (e.g. the patient could have kidney failure, probability of 60%, but the patient could also have liver failure, probability of 55%)

Table 2: Adapted MLM to Agent map.

The agent's beliefs capture and encode the information attributes of the guideline, the agent's plans capture and encode the deliberative and action attributes of the guideline, and the desire captures and encodes the motivational attributes of the guideline. This can be accurately and efficiently completed by using an adapted MLM to agent map as shown in Table 2, which was used in a proof of concept agent application developed by the authors.

The extended MLM structure provides a means to extract the guideline motivational components in the form of achieve, maintain and query goals. These motivational components provide the driving force behind the agent's activity. Using this approach an agent can act faithfully and autonomously on behalf of the guideline in a self-contained capacity. Thus when patient specific information is presented to the individual agents, via message passing, they have the ability to apply their encoded knowledge and logic, and provide a supportive response based solely on that information. Using this operation dynamic an agent module can also make use of the maximum supportive response from the other separate agent's (which represent guidelines) based on the known facts about the individual patient. By providing a framework which allows separate agents broadcast supportive communications to each other, the agent approach offers the opportunity for the data to be validated in a patient-centred fashion.

In summary, an ASK-agent is a self-contained software module which specialises in a specific condition, organ or disease affecting a particular population cohort. Other agents (software or human) interface with this software knowledge in light of the known patient information, and receive supportive information from the agent to aid in their patient specific healthcare delivery.

3 PROCEDURE TO CONVERT A GUIDELINE TO ASK-AGENT.

To provide a consistent technique for developing ASK-agent modules, which can act faithfully and accurately on behalf of a guideline, it is important to document, and explain the conversion process. The development of an ASK-agent module from a specific narrative guideline is based on the cycle shown in Figure 1. This procedure uses the expanded MLM mapping to the Agent Definition File (ADF) Extended Mark-up Language (XML) file and Java plans. The procedure is divided into six steps each of which is described in the following subsections.

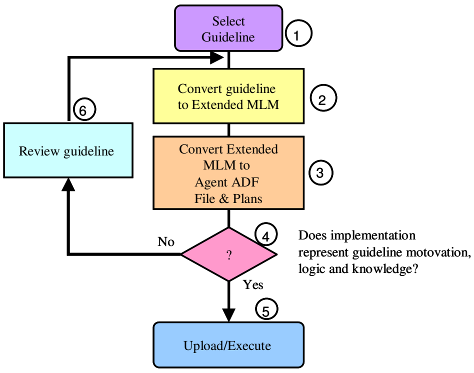

Figure 1: Encoding of expert agent procedure.

3.1 Select Guideline

The medical institution chooses a guideline that matches and complements their core activities. The types of guidelines being discussed are text based documents with conclusions being derived using rule-based or statistic-based decisions or numerical analysis. Neither the proposed ASK-agent nor existing CIG approaches can manage guidelines which have been developed using graphical imagery and charts.

3.2 Convert guideline to Extended MLM.

The guideline document is an amalgamation of a number of workflow activities which are triggered using motivational goals. Consider for illustration purposes the example of a generalised guideline extract shown in Table 2.

Evoke:
Arrival of laboratory results for analyte results X, Y and Z.
 	
Data:
PatientAge
PatientGender
 	
Logic:
IF (25Years < PatientAge >50Years) AND (40U/L < X > 130U/L) THEN perform *ActivityA*.
IF (PatientGender = Male AND (8U/L < Y > 40U/L) THEN perform *ActivityB*.
IF (PatientGender = Female AND (6U/L < Y > 35U/L) THEN perform *ActivityB*.
IF (PatientGender = Female AND (50U/L > Z) THEN perform *ActivityC*.
 	
Action:
ActivityA
ActivityB
ActivityC

Table 2: Generalised guideline extract.

The *evoke* slot relates to the triggering condition which is used to start this specific workflow path. The *data* information is retrieved from the Laboratory Information System (LIS) by completing an SQL query, or by executing an archetype query. An archetype is a recently employed term used in medical informatics. It is described in the CEN pre-standard prEN13606:2004(E) as a reusable, template model which presents a specific viewpoint of a domain reference model. The *logic* relates to a rule for selecting an activity. Finally the action is the activity to be performed, such as "print message to screen" or perform another activity.

3.3 Convert MLM into Agent ADF File and Java Plans.

The agent is encoded by directly mapping the extended MLM components to the ASK-agent. Code 1 illustrates how the Jadex ADF is encoded using XML. This file contains MLM slot components such as *data, evoke, action, logic, query goal, achieve goal, maintain goal* and *meta-level priority*. Meta-level reasoning is where two or more plans or goals can be triggered at the same time and a choice is needed on which one takes priority. For example consider the generalised guideline extract shown in Table 2. The first logical statement may be triggered calling on *activityA* to be executed. But any of the other three logic statements could also be triggered simultaneously calling on *activityB* or *activityC*. Do all need to be performed together, or should one activity be completed first? These decisions are traditionally realised centrally within the CIG engine, and are not part of the distributed MLM modules themselves. This motivation must be extracted from the original narrative guideline document and added to the BDI, providing the motivating force behind the agent's activity. Using Jadex this ordering of activities can be completed using priority attributes. Other components utilised by the expert agent model but not explicitly declared in either the narrative guideline or MLM are language, ontology and service descriptions. If the guideline was developed using a specific language or ontology, it is important that the sender of messages to it are aware of this, so a richer form of communication can be established. To this end the Jadex agent permits the declaration of an agent's language and ontology. The service description is used to register services the guideline can perform. In practice a guideline can perform many services, for example validate analyte Y and Z results if the *PatientAge* is known, or validate analyte X and Z results if *PatientGender* is known. These are services the ASK-agent is able to provide and are described using the Service Description component. For each service described using the ASK-agent eight parameters are used to describe it. These parameters are *name*, *type*, *ownership*, *GuidelineReference*, *InformationNeeded ValidationType*, *ontology* and *protocol*. These services are registered with the DF *goldenpages* agent. *Type* is used to identify the field of expertise to which the ASK-agent belongs (e.g. liver, kidney, and haematology). *Ownership* is used to identify the guideline implementation owner (e.g. department). *GuidelineReference* is used to identify the guideline that the ASK-agent represents (e.g. ISBN). *InformationNeeded* is a list of the data the ASK-agent requires in order to process the validation request (e.g. SerumALK_P, SerumALT, SerumGGT). The *InformationNeeded* titles can be developed to match the medical domain titles. Both ontology and protocol where described earlier. There presence in the service description is so other ASK-agents can identify the ontology and protocol used by this particular agent.

```
<!-- First Heading -->
<!-- <H3> Second Heading</H3>
     <H4> Third Heading</H4> -->
<agent xmlns:xsi="http…>
<imports>…</imports>
<capabilities>…</capabilities>
<beliefs>Data_MLM_Slot </beliefs>
   <goals>Evoke_MLM_Slot, Query
Goals, Achieve Goals, Maintain
Goals and Meta-level priority
</goals>
precondition, conditions and
triggers activating the goal are
represented by the Logic MLM
Slot
<plans>Action_MLM_Slot, Meta-
level priority </plans>
precondition, conditions and
triggers activating the plans are
represented by the Logic MLM
Slot
<events>…</events>
<languages>…</languages>
<ontologies>…</ontologies>
<expression>…</expression>
<servicedescription>…
</servicedescription>
   </agent>
```

Code 1: Jadex agent ADF XML constructor.

3.4 Expert agent testing.

Once the Agent ADF XML file and associated plans are developed the implementation needs to be thoroughly tested to ensure it represents the original guideline's motivations, logic and knowledge. Guidelines are autonomous self-contained entities; therefore the agent must act in a similar fashion. If the implementation of the guideline achieves this then it is uploaded into the main agent platform and can be used as part of the ASK-agent group.

3.5 Upload and execute agent module.

The developed ADF XML file and Java plans which represent the guideline are loaded into the JADE Remote Agent Management. Using the JADE Remote Agent Management GUI each agent is selected in turn using the "*Start New Agent*" button and loaded into the manager. Once installed the agent is able to act as part of the social group. The uploaded agents can be switched on, off or suspended by using the JADE Remote Agent Management GUI.

3.6 Review guideline conversion to ASK-agent.

If the agent module autonomous action does not represent the intended operation and understanding of the original narrative guideline, its encoding should be reviewed. This review continues until the agent represents the true operation of the guideline.

4 CONCLUSION

The purpose of this research is not to dismiss the CIG approach, but re-examine it from a processing point of view. This research demonstrates the agency approach offers a solution to the management of medical guidelines electronically, in a manner similar to that provided by the original narrative guidelines themselves. This is because of the synergy between the knowledge base, plans, decisions, action, goals and the self-contained components between guidelines and agents. It illustrates that this method permits the guideline content to be stored within one standalone agent module, and the entire knowledge, logic and data can be interfaced with by other agents when necessary. The ASK-agent approach permits a single guideline to be expressed within a standalone agent, but for true patient-centred validation to occur these separate agents must be able to communicate with other modules and collaborate. These last two aspects are covered in more detail in separate papers presented at this conference. So in summary the key differences between the CIG and ASK-agent approaches are detailed in Table 3.

Aspect	***CIG***	***ASK-Agent***
Processing	Centralised	Distributed
Guideline content	Fragmented within CIG engine.	Integrated as a standalone module.
Method of accessing information	Data entry and the execution of CIG rules.	Message passing.
Encoding of guideline content	Menu or/and graphical.	XML and Java.
Size of file(s) created	For a single guideline the CIG engine is small. However, as more MLM's are added the CIG engine application file becomes very large. Processing cannot be distributed.	For a single guideline the ASK-agent is larger than the CIG. However, as model is distributed the system file size is not a limitation

Table 3: Generalised guideline extract.

ACKNOWLEDGEMENTS

The authors wish to acknowledge and thank the work undertaken by developers of the Jadex agent platform at Distributed Systems and Information Systems Group, Computer Science Department, University of Hamburg. http://vsis-www.informatik.uni-hamburg.de/projects/jadex/

REFERENCES

Browne E.D., 2005. *Workflow Modelling of Coordinated Inter-Health-Provider Care Plans*, School of Computer and Information Science, University of South Australia, Adelaide, PhD Thesis January 2005.

Oosterhuis W.P., Bruns D.E., Watine J., Sandberg S., Horvath A. 2004. *Evidence-Based Guidelines in Laboratory Medicine: Principles and Methods*, published in Clin Chem, year 2004, Vol50 pgs 806-818.

Peleg M., Tu S., Bury J., Ciccarese P., Fox J., Greenes R., Hall R., Johnson P., Jones N., Kumar A., Miksch S., Quaglini, S., Shortliffe E, Stefanelli M. 2003. *Comparing Computer-interpretable Guideline Models: A Case-study Approach*, accessed via Journal of the American Medical Informatics Ass. Volume 10 Number 1 Jan / Feb 2003.

Rao A., Georgeff M., 1995. *BDI Agents: From Theory to Practice,* proceedings of the first international conference on multi agent systems (ICMMAS-95) San Francisco USA 1995.