

A Review of Decision Support Formats with Respect to Therapeutic Guidelines Limited Requirements

Dennis Wollersheim, PhD Student
Dept. of Computer Science and Computer Engineering, La Trobe University

Abstract

The National Health Information Advisory Council (NHIMAC) has recently suggested that the key stakeholders in the Australian healthcare scene gain agreement on a generic decision support language, considering the following candidates (NHIMAC 2001):

- Arden Syntax,
- Guideline Interchange Format (GLIF), and
- Proforma (UK).

This paper evaluates the suitability of these and other formats for the guidelines produced by Therapeutic Guidelines Ltd (TGL), as one national information resource. This is done through a qualitative comparison of guideline encodings.

Introduction: Medical experts, such as the family GP, are increasingly expected to *always* make the *best* decision. This is difficult. The amount of medical information in the world is increasing. Human brain capacity is not. Computers have the potential to help deal with all this information. In a general sense, computers are better than humans at managing masses of information and attendant complexity.

Guidelines have the potential to make medical practice more consistent, efficacious, safer and cost effective (Woolf et al, 2000). However, in practice, guidelines are often not used (Feder et al, 1999). Currently, most guidelines are in text format. On a micro level, computer implemented guidelines (CIG) have the potential to be more well used, by:

- improving accessibility, especially if integrated into prescribing system,
- allowing access to information with less loss of face than ‘looking it up in the book’,
- performing useful work, such as prescription printing,
- warning about potential hazards, and
- providing alternative ways to look at the information.

On a larger scale, CIG based systems can perform functions of recommendation, presentation, documentation, registration, communication, explanation, calculation, and aggregation.

This paper first describes the general CIG features, and describes the TGL environment as it relates to CIG implementation. Finally, CIG are compared, noting their applicability to the TGL situation.

Representation: On one hand, CIG representation is a moot point. It is merely a way to store and retrieve guideline content. But representation determines which content is captured, how it is stored, and how it is used. This, in turn, determines the interaction with clinicians, and guideline authors.

Authoring narrative guidelines is not easy, but at least human language is relatively standard, and sharable between readers. Authoring for CIG is even more onerous, because of the detail needed. As

well, computer readable guidelines cannot necessarily be shared between systems. This points out the desirability to implement the guidelines in a standard, sharable format, so that work done is maximally useful.

CIG Structure: It is an exciting time for CIGs. The field is new, and there is much development going on. Systems are exploring the boundaries of what is possible. At its most basic, a CIG takes the text of a narrative form guideline, and displays it on the computer screen. At the other extreme are algorithmic pathways designed to traverse a guideline from beginning to end.

Narrative guidelines have structure; headings, highlighting, table of contents and indexes each have (possibly implicit) meaning, and are useable. Descriptive CIGs extend this concept, further identifying and classifying elements of existing narrative guidelines, often using XML. Examples of such XML elements are *condition*, *recommendation*, *author*, and *level of evidence*. Descriptive representations are useful for purposes such as checklists during guideline authoring, advanced indexing, or stepping stones to more algorithmic representations. They are generally not executable, due to incomplete specification of element content. While descriptive CIG usefully adds structure to narrative guidelines, they have little current application.

Algorithmic representations are strict. They are designed to be computable. This means defined entry points, execution paths, and methods of integration with patient data. Their strength is the ability to take a set of inputs and make a recommendation. This makes them more provable and testable, and probably safer, but their rigidity makes them less useful in many situations. They are generally not browsable, and do not allow entry at an arbitrary point. However, “best clinical practice” is not always so certain or defined.

Expressivity: Expressivity is the ability of a CIG representation to accommodate use and content of guidelines. Expressivity is determined by structure, and feature set. A minimum algorithmic feature set would have a structure to hold patient state information, and an ability to make decisions, and to perform actions (Wang 2001). A more expansive representation includes such constructs as (Pattison-Gordon 1996) (Wang 2001):

Flow Control:

- temporal sequencing of actions: either one at a time, all at once, or no specific sequence,
- ability to perform loops, time delays, and secondary guideline invocation,
- multiple entry points,
- optional actions (actions that are only suggested),

Decisions:

- decision logic (probability / logic engineering / argumentation),
- computable eligibility criteria and decision logic, able to reference patient data items and perform boolean operations, arithmetic expressions, analysis of accumulated patient data, and operations involving time,

Actions:

- computable action specifications – for example, compute the appropriate amount of drug for a certain weight person,
- distinction between an instantaneous action and an ongoing activity,

Data:

- a (preferably standards based) patient data model, with the ability to unambiguously retrieve a specific temporal measurement, and specify a measure format (e.g. ‘high’ or 10),

- a structured vocabulary – allows cataloguing and location of items in a hierarchy, and a way to specify classification attributes, definitional attributes, and range of possible values,

Other:

- a patient description,
- goals and intent of actions,
- guideline intent – useful for guideline comparison, and determining granularity, and
- a way to specify didactic material.

Comprehensibility / complexity management: The complexity of medical decision making is massive, parallel to the complexity of the organism. A complete CIG would need to accommodate what is known about repair of this organism, and the knowledge is both deep (detailed, multi-factorial) and wide (across conditions, drugs, body areas). Because of this, CIG complexity escalates rapidly. Tools of data abstraction, data hiding, and modularity facilitate the management of complexity while retaining comprehensibility.

Knowledge Acquisition: The choice of representation partly determines the knowledge acquisition process. Knowledge acquisition is especially difficult in this domain, due to the complexity of the task, user unfamiliarity with both the tools and the task, and the relative inexperience of the (normally domain expert) user. These factors increase the difficulty of developing an appropriate interface. For example, the process of writing, an earlier version of knowledge acquisition, has evolved a relatively standard interface: language. But this evolution was not instantaneous. The process of developing a good authoring interface for CIG knowledge acquisition will be iterative.

Secondly, the knowledge captured is based on a shifting foundation. Medicine is constantly changing. Terminology changes, and more information is discovered. This will demand a system that can change accordingly. As in human beings, the most important faculty here is the ability to learn and adapt.

An effective, long lasting system will need to be flexible. It will be modified many times in its life. Complex systems are difficult to modify without introducing errors. Ontologies can help with this.

Ontologies: An ontology is a classification scheme that defines essential properties and relations of objects. In the CIG domain, these objects can be guideline elements, like medical conditions, procedures, drugs, or even guideline structural components. Ontologies can provide a facility for data abstraction, and a way to separate the knowledge base from the execution engine. This means that relationships can be defined once, and reused elsewhere. Ontologies facilitate both prototyping, by isolating effects of changes, and verification, by making the rules and data elements explicit.

For example, if a rule in a knowledge base is a prohibition against prescribing a drug of type X in the presence of an event of type Y, it is better to have external lists of all types X and Y than to make the guideline author enumerate them all. An ontology would contain these lists. Another example is, “there is a class of things called drugs, and a sub-class called beta-lactams, and an instance called amoxicillin”, and then be able to answer the question, ‘is amoxicillin a beta-lactam?’

Safety: Medical CIG has a critical responsibility. Safety is important. Aspects of safety include:

- provability – can the representation be proven correct? This includes the issue of *completeness*, that is, not leaving anything out, covering all possibilities,
- testedness – how well tested is this representation, and associated tools? and

- ease of verification – this pertains to guideline authoring workflow; is it simple to verify correctness?

A system will be less likely to have errors if it:(Fox and Das 2000)

- is designed according to established software engineering principles, such as a component based design,
- separates knowledge and logic,
- authored with a simple toolkit,
- has a minimal feature set, but adequate to encode guidelines in a simple and straightforward manner,
- has straightforward decision making, and
- can use clinician knowledge, and be overridden.

Therapeutic Guidelines: The guidelines produced by Therapeutic Guidelines Limited (TGL) are an Australian national therapeutic information resource. TGL began publishing narrative guideline handbooks, and currently have a first generation CIG, consisting of narrative text in hyperlinked HTML pages.

TGL needs CIG. Its production streams are proliferating; CIG will add structure to the documents, streamlining multiple outputs, and facilitate error checking. The world is computerising; TGL wants guidelines bought and used, and is therefore motivated. From the other perspective, CIG can usefully employ TGL guidelines as a starting point. TGL information is concise and up to date. The research has been done, the experts consulted.

The Guidelines have evolved (continuing a rich ‘human language’ tradition) to fit into the clinicians workflow, fulfilling purposes such as browsing, specific problem solving, hazard checking, drug reference, and comparisons. They do this with tools such as a table of contents, a multi-level index, physical text layout, concise prose, and tables.

This evolution has come in response to pressure from clinicians and guidelines authors, indirectly answering the question “what decisions are we supporting?” The guidelines support a wide variety of decisions. This has necessitated a form that has many entry points, with an ability to ‘zoom in and out’, moving from overview to detail.

TGL aims for broad medical coverage, and dealing mainly with acute, single consultation issues. The shallow coverage implies low algorithmic requirements, requiring a relatively inexpressive knowledge representation.

Narrative Translation: Extraction of deterministic algorithmic elements from narrative guidelines is generally difficult. Full specification of condition, treatment and assessment details is needed (Pattison-Gordon and J. J. Cimino 1996). Narrative guidelines often:

- expect common sense,
- depend on physician base knowledge and assumptions,
- have no formal structure,
- imprecisely specify terms, flow control, and logic (Greenes, Peleg et al. 2001),
- miss explicit definitions of symptoms and adverse events,
- account insufficiently for co-morbidity, and

- lack detailed information on how to use the drugs recommended, (Goldstein, Hoffman et al. 2000)

Mechanics of translation: At one level, TGL guidelines can be modeled as a simple tree structure, following the text's hierarchal structure. For example, the tree above conjunctivitis in TG: Antibiotic is:

Eye infections → Conjunctival infections → Conjunctivitis

An algorithmic CIG could represent this by the set of decision steps 1) does the patient have an eye infection? 2) does the patient have a conjunctival infection?, and 3) does the patient have conjunctivitis?

The simplicity of the hierarchal overview belies the complexity of the narrative detail. The prose is indeterminate, relying on interpretation. The first paragraph of the conjunctivitis section follows:

*“Most mild conjunctivitis is allergic or irritative in origin. Infective causes **may** be viral or bacterial and are **often** difficult to distinguish clinically from allergic or irritative manifestations. Empirical antibiotic treatment **may** therefore be appropriate.” (Writing Group for Therapeutic Guidelines Antibiotic 2000), (emphasis added)*

Examples of indeterminacy are indicated by the words *most*, *often*, and *may*. An algorithmic CIG needs to know which, and under precisely what circumstances, ‘empirical antibiotic treatment’ is necessary.

A confounding issue is the TGL writing process. The content is written by a committee, using consensus. Turning the above paragraph into a set of definite rules will require that the committee understand algorithmic CIG implementation, merely so that they will know what they are agreeing to. As well, they will have to make more definite decisions, using precise language, which is particularly difficult when the evidence is incomplete and under the constraint of consensus.

A third difficulty has to do with the nascent state of CIG authoring. The narrative form has been around for a long time, and has agreed meaning. CIG implementation is in a state of flux. It is difficult for authors to take ownership of content where the representation is not fixed.

Methods: A section of the TGL guidelines were implemented in CIG formats: Arden (Jenders 2000), Proforma (Fox, Johns et al. 1998), HGML (Hagerty, Pickens et al. 2000), and GEM (Shiffman, Karras et al. 2000). The resulting implementation was checked by TGL guideline editors. These representations, along with that of Prodigy (Sugden, Purves et al. 1999) and GLIF (Peleg, Boxwala et al. 2000; Peleg, Boxwala et al. 2001), were reviewed, noting design features and usability issues.

Results: Hypertext Graphic Markup Language (HGML) is a XML based descriptive representation, exemplified in figure 1. Its chief advantage is that it would easily fit into the TGL production cycle, being an in-situ markup of existing guideline text. Using five XML elements, it performs the basic task of identifying and connecting statements, conditions and recommendations. But the simplicity is limiting. For example, there is no way of encoding boolean conjunctions.

```

<statement> <condition> Patients with conjunctivitis who have
    <variable>significant pain</variable> ,
    <variable>loss of vision</variable> or
    <variable>photophobia</variable>
</condition>
require
<recommendation>prompt referral to an ophthalmologist</recommendation> </statement>

```

Figure 1: TG: Antibiotic Conjunctivitis example encoded in HGML

GEM: Guideline Elements Markup (GEM) is another XML based descriptive representation. In contrast to HGML, it is not simple, with greater than 100 element types. The representation is even expressive enough to hold the algorithmic guideline format GLIF.

GEM, like the algorithmic representations, has no link between the extracted knowledge and the original document. This division means that content does not have to physically exist in an original document (it can be implied and translated), but, revision is made more difficult.

Descriptive representations are closer to the current state of the TGL production process. Their structure itself is useful; the exercise of using them helps authors think about guidelines in a more ordered way. They are potentially useful for consistency, error checking and indexing, and even the absence of values in slots will show what algorithmic details are needed. But in and of themselves, descriptive representations are not very useful yet.

Arden: Arden is one of the original algorithmic CIG decision support languages. It was originally implemented to work as ‘watchdogs’ in hospital situations, simple programs watching data values.

Arden is not useful for TGL guidelines. In format, Arden is a text based programming language, and would be difficult for domain experts to use. Arden’s simple grammar makes individual modules weak. More intricate guidelines can be constructed via multiple modules, but this is not ideal as complexity increases and comprehensibility plummets.

GLIF: GLIF (version 3) is another algorithmic format, was designed by a consortium of US academics to promote guideline sharing. As such, it includes features from many sources, and is very expressive. This full feature set invites complexity, and in response to that, they incorporate a range of complexity management features, such as data abstraction, data hiding, and modularity. Because of its wide development process, it is supported, and is becoming a standard. Authoring tools exist in an alpha stage.

Proforma: Proforma is an algorithmic format designed with a focus on safety. It is used at several sites in the UK. The object oriented knowledge representation is simple and powerful. Proforma’s decision making process is notable in that it uses logic of argumentation.

Proforma and GLIF are very expressive, and have the ability to implement a broad range of guidelines. TGL requirements are easily met by either of them. But, in fact, this high expressivity is a burden. Decreased comprehensibility and increased domain expert authoring difficulty, multiplied by the broad TGL guideline base, would make production a nightmare.

Secondly, like GEM, these algorithmic representations have no way of connecting chunks of narrative text with related algorithmic CIG content. This increases the distance between the two production

streams. Because algorithmic CIG is an active system, it is likely to be more interconnected than narrative text. Tracking ramifications of changes in a large guideline base such as TGL would be difficult.

A final point regarding algorithmic guidelines is their usability. The goal for TGL CIG is a decision support tool that will unobtrusively troubleshoot prescribing. They want the doctor to update their knowledge, reflect on their practice, and facilitate the information exchange between doctor and patient, and on the whole, be empowered by the interaction. Narrative guidelines have this functionality. This is not the case with current generation algorithmic guidelines. Their structure is too rigid: asking of questions, giving of answers. This deskills the clinician, and does not make use of their thinking. While algorithmic CIG is useful for high risk or novel decisions, they only perform part of current TGL function.

Prodigy: Most of the above systems were developed from a research perspective. This is valuable, but actual implementation teaches other lessons. The Prodigy project, a UK government initiative, is such a case. Prodigy is mandated to put CIG on clinician's desktops. They have made usability as a high priority, and succeeded in implementing that within an executable algorithmic structure.

The first step to this end was to have the system integrate with existing vendor software. They spent the majority of their first budget developing relationships with the vendors. Another important relationship was that of doctor to software. They noted the widespread paucity of valid patient data, and built a system that degrades gracefully; less patient information leads merely to an expansion of choice. When more information is present, clinicians are rewarded by a shorter pick list. In a similar fashion, the user is always free to choose any guidance, but the system offers it's best guess.

Discussion: TGL has broader goals than just making an executable guideline. They want the system that will be used. That means that they have to fit in with clinician requirements and workflow. They are trying build the doctor software relationship, and use this relationship to modify behaviour. This is a delicate task. A system that does this will provide useful tools, be completely integrated with patient and drug data sources, offer optional structure wherever possible, and reward good data capture habits. It will permit freedom to choose and browse.

TGL is not ready for full fledged algorithmic CIG; the leap would be too great. But more importantly, algorithmic CIG is possibly not the correct goal. This format may not engender desired usage, or build the desired relationship. Alternatively, descriptive CIG is a step forward from the current situation; it would build on existing structure, making the data more useable, and is amenable to an incremental approach.

References

Feder G, Eccles M, Grol R, Griffiths C, Grimshaw J. (1999) 'Clinical guidelines: using clinical guidelines'. *BMJ* 1999;318:728-30

Fox, J. and S. Das (2000). *Safe and Sound: Artificial Intelligence in Hazardous Applications*. Menlo Park, CA, American Association for Artificial Intelligence.

Fox, J., N. Johns, et al. (1998). 'Disseminating medical knowledge: the PROforma approach.' *Artif Intell Med* 14(1-2): 157-81.

- Goldstein, M. K., B. B. Hoffman, et al. (2000). *Implementing Clinical Practice Guidelines While Taking Account of Changing Evidence: ATHENA, an Easily Modifiable Decision-Support System for Managing Hypertension in Primary Care*. AMIA Annual Symposium, Los Angeles, CA.
- Greenes, R. A., M. Peleg, et al. (2001). 'Sharable Computer-Based Clinical Practice Guidelines: Rationale, Obstacles, Approaches, and Prospects.' *Medinfo*.
- Hagerty, C. G., D. Pickens, et al. (2000). *HGML: A Hypertext Guideline Markup Language*. Proc. Annual Meeting of the . Annual Meeting of the American Medical Informatics Association (AMIA), Los Angeles.
- Jenders, R. A. (2000). The Arden Syntax for Medical Logic Systems. 2000.
- NHIMAC (2001). Setting the Standards: a national health information standards plan for Australia, Commonwealth of Australia. 2001.
- Pattison-Gordon, E. and G. H. J. J. Cimino, S. W. Tu, J. H. Gennari, N. L. Jain, & R. A. Greenes (1996). Requirements of a Sharable Guideline Representation for Computer Applications, Stanford University.
- Peleg, M., A. A. Boxwala, et al. (2000). 'GLIF3: the evolution of a guideline representation format.' *Proc AMIA Symp*: 645-9.
- Peleg, M., A. A. Boxwala, et al. (2001). 'Handling Expressiveness and Comprehensibility Requirements in GLIF3.' *Medinfo*.
- Shiffman, R., B. Karras, et al. (2000). 'GEM: A proposal for a more comprehensive guideline document model using XML.' *JAMIA* 7(5): 488-498.
- Sugden, B., I. N. Purves, et al. (1999). The PRODIGY Knowledge Architecture Requirements for Chronic Disease Management in Primary Care, Stanford Medical Informatics.
- Wang, D., M. Peleg, et al. (2001). Representation of Clinical practice Guidelines. London, Medinfo.
- Woolf SH, Grol R, Hutchinson A, Eccles M, Grimshaw J. (1999) 'Clinical guidelines: potential benefits, limitations, and harms of clinical guidelines'. *BMJ* 1999;318:527-30
- Writing Group for Therapeutic Guidelines Antibiotic (2000). *Therapeutic guidelines: Antibiotic*. Melbourne, Therapeutic Guidelines Ltd.

Permissions/Disclosure of research funds

This paper describes research being carried out within projects funded by the Australian Research Council, National Prescribing Service, and Therapeutic Guidelines Limited

Acknowledgements

This project could not have been possible without the support and contributions of Dr. Wenny Rayahu, Dr. Bryn Lewis, Dr. Ken Harvey, Dr. Jonathan Dartnell, Assoc. Prof. Teng Liaw, and Ms. Elizabeth Deveny.

Contact details

Dennis Wollersheim, PhD Student. Department of Computer Science and Computer Engineering, La Trobe University. Bundoora, Australia. Email: dewoller@cs.latrobe.edu.au