A quality and safety framework for point-of-care clinical guidelines

John Fox PhD and Jonathan Bury MB ChB,
Advanced Computation Laboratory, Imperial Cancer Research Fund,
London, WC2A 3PX, United Kingdom.

ABSTRACT

The electronic dissemination of medical knowledge in the form of executable clinical guidelines and decision support systems must be accompanied by comprehensive methods for ensuring the quality of their knowledge content and their safety in use. This paper outlines a set of quality and safety requirements, and reviews three current guideline technologies, the Arden Syntax, GLIF and PROforma, against these requirements. The approaches used in these technologies have different strengths, and we propose a general framework for ensuring quality and safety that combines them. This framework brings together the normal documentation standards of medical publishing, rigorous design methods from software engineering, and active safety management techniques from artificial intelligence.

INTRODUCTION

A search for "clinical guidelines" on the World Wide Web will currently return hundreds of thousands of hits. Even allowing for many duplicates, there is clearly a huge worldwide interest in writing and publishing "best practice" clinical guidelines, motivated by the goal of encouraging evidence-based clinical practice. Yet despite the clear need to systematically review, document and disseminate best practice, there are reasons to doubt that this effort will produce the intended effect. Why, after all, should busy doctors find time to read guidelines on screen any more readily than on paper? And why should we feel confident that even when guidelines have been read, the knowledge they present will be properly internalised and correctly applied?

For these reasons there is growing interest in new ways of disseminating medical knowledge more effectively, notably by translating the content of guidelines from textual form into a logical form that can be executed or "enacted" by a computer, thereby providing patient-specific prompts, reminders and other forms of advice on an individual's care[1]. There is growing evidence of the effectiveness of such guidelines[2].

With this development, however, comes a new set of responsibilities. The authors of traditional textual guidelines commonly assume that the clinical user of the material will apply his or her clinical judgement in applying it, ensuring that a guideline is relevant to the context of use and adapting it to specific circumstances if appropriate. The use of machine-generated advice, however (particularly if it is not accompanied by general and/or patient-specific explanatory material) has the potential to introduce the risk of uncritical compliance with an inappropriate guideline, and even of creating hazards for the patient.

"OpenClinical"

We are developing an experimental web-based repository of enactable clinical guidelines called OpenClinical. The aim of our research is primarily to establish a sound publishing methodology for creating and disseminating guidelines in such a way that users can have confidence in the quality of the material and its safety in use. We believe that this methodology must address at least four requirements:

- Comprehensive on-line documentation to support quality and user review
- A systematic publishing life-cycle supporting the creation and maintenance of the medical content and its evidence base
- Rigorous software engineering methods to ensure application integrity and platform reliability
- Explicit safety and hazard management at run time

In what follows we briefly review how these issues have been addressed in three prominent guideline enactment technologies, and then bring together the various methods in a proposal for a general quality and safety framework.

STANDARDISED GUIDELINE FORMATS

It is widely accepted that standardised guideline representations and interchange formats will be needed if we are to create a substantial body of enactable guidelines and related material[1,3,4,5]. The motivations for creating standard formats include the need for interoperability of guidelines, reusability of standard components, readability and sharing of experience. The Arden Syntax, the Intermed Collaboratory's Guideline Interchange Format

1067—5027/00/$5.00 © 2000 AMIA, Inc.

(GLIF), and PRO*forma* have been proposed as means of achieving these objectives. However, they are technically quite distinct and have addressed the requirements above in different ways.

The Arden Syntax has been under development since the early 1990's. It provides a framework for the representation of single clinical decisions in the form of "Medical Logic Modules" (MLMs). MLMs are generally embedded within clinical information systems; input data are usually derived from a patient database, and the output is usually a message to health care providers, giving prompts, reminders and other simple decision aids. Methods for chaining together multiple MLMs to provide more complex decision support functions have been studied[3], but to date have had limited success.

GLIF (Guideline Interchange Format) is a guideline representation and sharing format developed by the Intermed Internet Collaboratory. It draws on the considerable experience of several established groups with 4 earlier representation systems, including the Arden Syntax. GLIF provides tools for more complex guideline representation than the Arden Syntax, including branching structures, workflow representations, a richer decision making capability and more complex action specifications[4].

PRO*forma* is a formal specification language for describing clinical procedures in terms of a small number of intuitive, generic "tasks" and their interrelationships. PRO*forma* supports formal models of decisions (e.g. prescribing and referral decisions), plans (e.g. care plans, clinical trial protocols), actions (e.g. clinical orders) and "enquiries" - actions that return information about a patient (e.g. through form filling or database queries). PRO*forma* also supports *composition* of complex clinical procedures from these components, and provides constructs for logical and time-based scheduling of clinical tasks.

QUALITY AND SAFETY MANAGEMENT

The Arden Syntax

The Arden Syntax has a well-structured approach to documentation within an MLM. Each MLM is associated with 'slots' for this purpose, grouped into 3 categories: maintenance, library and knowledge. The MLM definition specification includes 9 maintenance fields and 5 library fields. Three of the maintenance fields concern accountability: Who wrote the MLM? At what institution? Who is responsible for the local implementation of the MLM and its use? Others concern applicability: When was the MLM produced? Which version is this? To what extent has it been validated? The library fields not only address indexing and links to further information, including citations that may support or question the assumptions of the MLM, but also deal with its clinical role: What is its purpose? In what context is it appropriate?

A clear statement of the status of the medical knowledge content of the guideline is supported through the "validation" slot. This can have 1 of 4 values: "testing", "research", "production" and "expired". Although we are not aware of a formal evaluation and development lifecycle for the clinical knowledge content, the ASTM standard for the Arden Syntax gives guidance on the appropriate use of these terms.

GLIF

The GLIF format includes documentation concepts paralleling some of those used in the Arden Syntax, including the guideline name, a list of authors, a statement of the guideline's intention, a specification of patient eligibility criteria and a list of supporting didactic material. To date less quality control concepts are supported in the GLIF model, though this probably just reflects its relative immaturity and the forthcoming GLIF-3 language specification is expected to address this. However, whilst patient eligibility criteria are only implicitly represented within the free text library fields of "Purpose" and "Explanation" in the Arden Syntax, GLIF represents these criteria in explicit, machine interpretable terms.

GLIF is also more expressive than the Arden Syntax in that it can be used to define complex clinical procedures in terms of guideline "steps" and simple decisions. Complexity presents opportunities for more errors, however, so the process of guideline *authoring* is supported by graphical design tools for laying out the required steps in the guideline. These tools allow the author to lay out the steps in a form reminiscent of a flow diagram or clinical algorithm.

PRO*forma*

PRO*forma* is a *formal specification language*, comparable to languages such as Z and VDM that are used extensively in software engineering[6,7]. The motivations for developing formal design techniques have been to increase the clarity of a knowledge representation, to remove many of the apparently *ad hoc* practices associated with knowledge systems development, and to provide techniques for automated verification and validation of the system knowledge base[8].

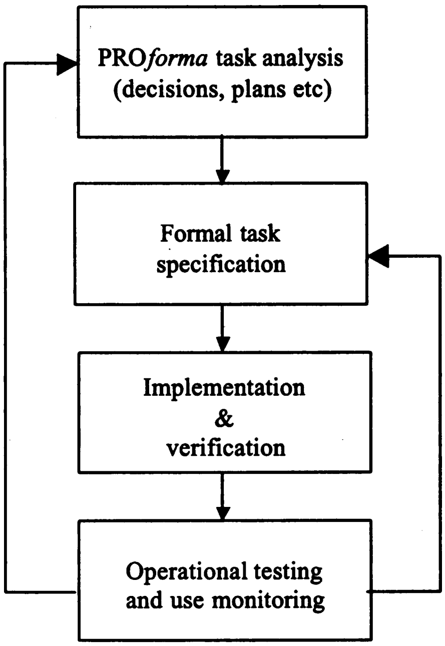
PRO*forma* task analysis (decisions, plans etc)
Formal task specification
Implementation & verification
Operational testing and use monitoring

Figure 1 : PRO*forma* technical design lifecycle

The core of the PRO*forma* guideline representation format is the set of generic ***task models*** – enquiries, decisions, plans and actions[1]. Tasks are formal software objects, that can be composed into networks representing plans or procedures that are carried out over time. Since we have a formal model of the general properties of decisions, plans and other tasks, and the relationships between them, there is scope for automatically carrying out extensive logical checks on the guideline specification, including checks for:

- syntactical errors
- incomplete or inconsistent task definitions
- data referred to in inference rules but not defined
- inconsistent constraints (e.g. scheduling or temporal constraints)

Since the language is logically well structured and formally defined it is also possible to build a set of Computer Assisted Software Engineering (CASE) tools for rigorously designing and verifying the consistency, completeness etc. of an application. Since PRO*forma* is a machine executable format, there is considerable scope for systematic testing. PRO*forma*'s major strength is the provision of a comprehensive set of CASE tools to support a systematic guideline technical development lifecycle, illustrated in figure 1.

The PRO*forma* language does not explicitly define documentation standards, though authoring tools would normally support at least basic documentation. For example, a PRO*forma* authoring environment that we have developed in collaboration with *Infer*Med Ltd provides a range of documentation and version control functions.

OpenClinical: A UNIFIED APPROACH TO QUALITY AND SAFETY

OpenClinical aims to facilitate the dissemination of machine-enactable guidelines and decision support applications, using the PRO*forma* format as its standard platform. To do this we have adopted the development life-cycle and the suite of software engineering tools referred to above, together with a comprehensive documentation standard based on that of Arden and GLIF.

However, there is still much to be done if we are to satisfy all the quality and safety requirements set out in the introduction. Even the most rigorous engineering and documentation procedures cannot ensure that the medical knowledge content of a guideline is correct, or that that medical knowledge is applied appropriately and safely in practice[9].

Knowledge content lifecycle

In our view we need to develop a comprehensive publishing methodology based on an ongoing systematic development and reappraisal process, paralleling that used by the pharmaceutical industry in developing and testing new drugs. Our proposed knowledge content life-cycle is illustrated by the "figure of eight" model shown in figure 2. This life-cycle starts at the point at which the authors of systematic reviews of clinical research normally finish – the published text guideline. Working from this source we use the PRO*forma* authoring tools to implement the guideline in a form that is machine enactable, as summarised in figure 1. The guideline is first appraised "off-line", by testing it against records of past cases[10] or in simulated but realistic conditions[11].

Once off-line testing has shown that the guideline is logically sound it can enter a second phase of testing - a controlled clinical trial. A guideline is like any other clinical procedure and is therefore subject to assessment against the "normal" practice as a control condition. Since the PRO*forma* language is also used as the representation language for *Infer*Med's MACRO clinical trial system (www.infermed.com), this software can be used directly to carry out a clinical study in which the guideline doubles as trial protocol and guideline implementation language. If

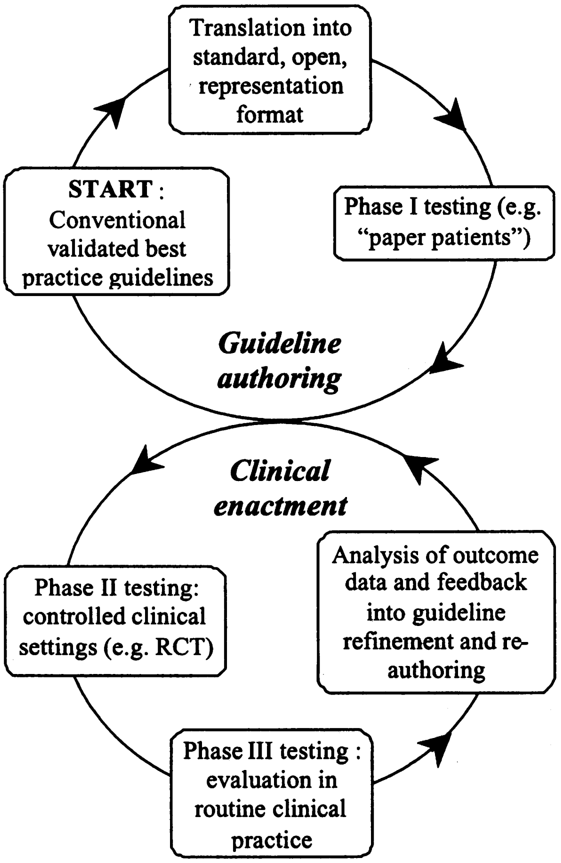
Translation into standard, open, representation format
Phase I testing (e.g. "paper patients")
START : Conventional validated best practice guidelines
Guideline authoring
Clinical enactment
Analysis of outcome data and feedback into guideline refinement and re-authoring
Phase II testing: controlled clinical settings (e.g. RCT)
Phase III testing : evaluation in routine clinical practice

Figure 2 : A methodology for ensuring the quality of the clinical knowledge content of enactable electronic clinical guidelines.

this trial is satisfactory then the guideline can be tested in routine clinical use (phase III), either as an embedded application or via a standard browser over the web (as is the case in OpenClinical). Finally, since the guideline enactment engine can also be used to record data about clinical use and outcomes we can maintain a database for recording comprehensive data for subsequent statistical analysis and data mining. The results of this analysis can be fed back into the next cycle of the literature review for creating the next generation of guideline. This authoring and enactment life-cycle addresses the issue of the *quality of the medical knowledge* in a guideline, which will presumably have a direct bearing on its efficacy.

As with drugs and other medical technologies, ensuring the *efficacy* of a guideline does not necessarily ensure *safety*. Accordingly, a major topic of our current research concerns how the runtime engine that enacts a guideline can incorporate functions for active management of patient safety during clinical use. Predicting in advance, and explicitly planning for, the entire range of hazardous situations that may occur during guideline enactment is unlikely to be feasible. PRO*forma*'s ability to simultaneously execute multiple plans or software agents enables the deployment of plans which continuously monitor for adverse events or hazardous trends and provide appropriate warnings. By providing such "Guardian Agents" with the facility to predict the consequences of proposed actions in the context of the current patient state, using external knowledge bases and first-order logic functions, it should be possible to avoid or respond appropriately to hazards unforeseen at the time of guideline authoring. We are also exploring techniques for the automatic dynamic modification and generation of plans, to enable substitute or additional clinical actions to be proposed in order to correct or prevent the hazard. This work is still incomplete but very promising results are described in detail elsewhere[9].

CONCLUSION

We have proposed a quality and safety methodology for developing and using clinical guidelines that addresses four distinct requirements:

- A documentation system for decision support software that informs the user of its medical and technical validity and origins, and provides links to further information
- Use of a formal guideline representation that facilitates automatic verification and checking of the consistency and completeness of the guideline definition
- An evidence-based lifecycle for creating and validating the medical content of the guideline
- The development of software capable of supporting active safety management and dynamic management of unforeseen hazards, including interactions between guidelines

Whilst the first three of these approaches are all in use in different systems, there has been less work on software capable of active safety management as well as rigorous quality control. Initial results are promising, however, and if these are confirmed the method will be adopted as part of the OpenClinical publishing methodology.

ACKNOWLEDGEMENTS

We thank Dr Aziz Boxwala for his help in preparing this manuscript.

REFERENCES

1. Fox J, Johns N, Rahmanzadeh A. Disseminating Medical Knowledge - The PRO*forma* Approach. Artificial Intelligence in Medicine, 1998; 14: 157-181
2. Johnston ME, Langton KB, Haynes RB, Mathieu A. Effects of computer-based clinical decision support systems on clinician performance and patient outcome. A critical appraisal of research. Ann Intern Med, 1994; 120(2): 135-42
3. Hripcsak G, Ludemann P, Pyror TA, Wigertz OB, Clayton PD. Rationale for the Arden Syntax. Comput Biomed Res, 1994; 27: 291-324
4. Sherman EH, Hripcsak G, Starren J, Jenders RA, Clayton P. Using intermediate states to improve the ability of the Arden Syntax to implement care plans and reuse knowledge. Proc Annu Symp Comput Appl Med Care. 1995;: 238-42
5. Ohno-Machado L, Gennari JH, Murphy SN, et al. The guideline interchange format: a model for representing guidelines. J Am Med Inform Assoc, 1998; 5(4): 357-72
6. Spivey, JM. The Z notation: a reference manual. New York: Prentice-Hall, 1989
7. Jones C. Systematic software development using VDM. London: Prentice-Hall International, 1986
8. Van Harmelen F, Balder JR. $(ML)^2$: A formal language for KADS models of expertise. Knowledge Acquisition, 1992; 4(1): 127-161
9. Fox J, Das S. Safe and sound : artificial intelligence in hazardous applications. American Association for Artificial Intelligence and MIT Press, Cambridge: MIT, 2000
10. Walton RT, Gierl C, Yudkin P, Mistry H, Vessey MP, Fox J. Evaluation of computer support for prescribing (CAPSULE) using simulated cases. BMJ, 1997; 315(7111): 791-795
11. Emery J, Walton R, Coulson A, Glasspool DW, Ziebland S, Fox J. Computer support for recording and interpreting family histories of breast and ovarian cancer in primary care (RAGs): qualitative evaluation with simulated patients. BMJ, 1999; 319(7201): 32-36