

PROforma: A Method And Language For Specifying Clinical Guidelines And Protocols

John FOX ¹, Nicky JOHNS, Ali RAHMANZADEH, Richard THOMSON
Imperial Cancer Research Fund, Lincoln's Inn Fields, London, WC2A 3PX, UK
¹*tel +44 171 269 3624, email jf@acli.net.uk*

Abstract. The need for flexible and well understood knowledge representations which are capable of capturing clinical guidelines and protocols for decision support systems is widely recognised. PROforma is a method for specifying clinical guidelines and protocols in a form which can be executed by a computer to support the management of medical procedures and clinical decision making. The PROforma method comprises a graphical notation for designing protocols and a formal knowledge representation language to enable clinical concepts to be interpreted by a computer. This paper provides an overview of the motivation, structure and use of PROforma in developing computerised clinical guidelines.

Keywords: Protocol-based care, decision support systems, safety, knowledge representation, knowledge based systems.

1. Introduction

Various attempts have been made to define knowledge representation languages capable of capturing clinical procedures as knowledge bases. These include the "Arden syntax", a Pascal-like language intended to provide a standard format in which medical knowledge can be specified and shared. However, the available languages are widely thought to be insufficiently general, versatile or theoretically sound for general use. For example, as Musen et al note [1], the developers of the Arden syntax themselves recognise that "this new standard has significant limitations: the language currently supports only atomic data types, lacks a defined semantics for making temporal comparisons or for performing data abstraction, and provides no principled way to represent clinical guidelines that are more complex than individual situation-action rules" [1]. PROforma is being developed to address these limitations and to satisfy a number of broad requirements: clinical generality, medical expressiveness, operational flexibility, soundness and safety, verifiability and support for reusability of knowledge bases.

PROforma is a formal specification language in the sense used in software engineering: a formalism whose meaning is well understood and which can consequently be exploited in ensuring quality and safety of applications. It is also a knowledge representation language in that it is structured around a set of concepts which are intelligible to clinical professionals yet facilitate sound design and safe automation of clinical procedures.

Three kinds of knowledge are used in combination when making patient management decisions: knowledge of the specific patient (personal details, problems, symptoms, current medications and so forth), general medical knowledge (of diseases, symptoms, tests, drugs), and knowledge of medical procedures (what should be done when). PROforma is primarily focused on the last type of knowledge; our intention is that PROforma should be able to accommodate different medical knowledge models and patient data models.

2. Computerising a clinical procedure with PROforma

Preparation of a PROforma application (e.g. a computerised care guideline or clinical protocol) is a two-step process. In step 1, a high level description of the guideline is developed using a graphical design package. (The source for this is typically a published document which sets out the guideline, ideally from an authoritative body.) The resulting diagram shows the main clinical tasks required in an easily understood form, including their logical and temporal interrelationships. In step 2, the graphical structure is converted into a database, using a knowledge editing tool, and populated with the detailed procedural and medical knowledge base required to execute the guideline.

2.1. Representing the high level structure of a guideline in graphical form

The top level structure in PROforma is the *task*. PROforma applications are designed to provide support for a comprehensive range of clinical tasks, such as decision making, scheduling of actions over time, reminders for patient data collection, monitoring adverse

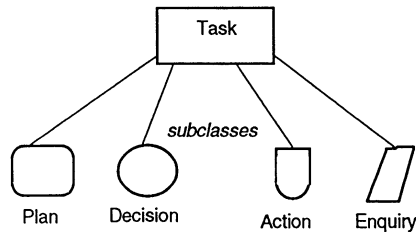


Figure 1: The PROforma task ontology

events and so forth. Starting from the available knowledge sources, an application designer typically develops the graphical network which summarises the decisions, actions and other tasks recommended by the guideline. PROforma supports four basic classes of task (figure 1):

- A *plan* is a sequence of subtasks, or components, which need to be carried out to achieve a clinical (e.g. therapeutic) objective. Plan components are usually ordered, to reflect temporal, logical, resource or other constraints. Clinical guidelines and protocols are plans, but are slightly special because they represent self-contained procedures with specific clinical objectives.
- A *decision* task, such as a diagnosis or therapy decision, is modelled as a set of decision options, rules for arguing for and against the options, and “commitment rules” which determine when the decision should be made.
- An *action* task is a procedure which is “external” to the computer system, such as the task of administering an injection.
- An *enquiry* is a task whose objective is to obtain an item of information which is needed in order to complete a procedure or take a decision. The specification of an enquiry includes a description of the information required (e.g. a lab. result) and a method for getting it (e.g. by a query on a local patient record, or a remote laboratory database).

Figure 2 shows a schematic view of a protocol in the PROforma graphical notation. This figure provides a simplified view of a task network representing a protocol (FB1) used at the Institut Bergonié for the treatment of limited breast cancer. Tasks are represented by simple icons, and the relationships between them by arrows and other graphical devices (circles

represent decisions, rectangles represent plans, and arrows represent scheduling constraints). A procedure which lies inside another procedure forms a component of that procedure, but may also potentially be reused in other protocols. Tasks can be repeated or cycled (e.g. cmf therapy in figure 1).

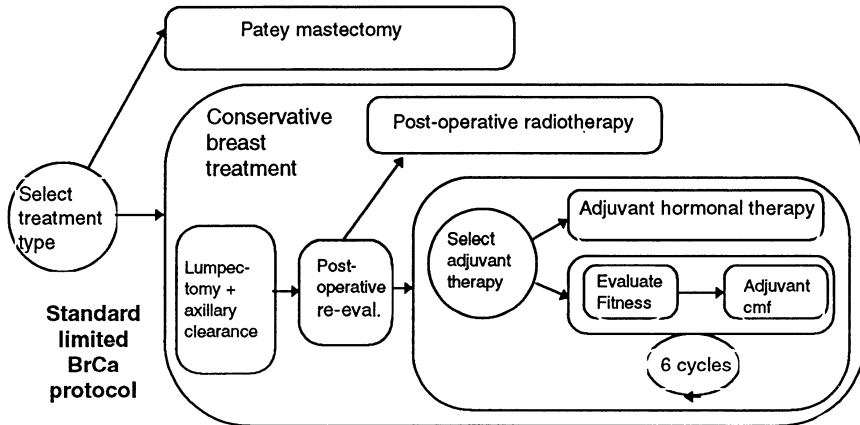


Figure 2: Simplified task analysis for FB1 combined therapy protocol for treatment of breast cancer. Examples of plans and decisions are shown here.

This high level structure shows that the first task to be carried out in FB1 is a decision about the appropriate treatment type for the patient. Either Patey mastectomy may be chosen or, shown here in detail, conservative treatment. Conservative treatment decomposes into a sequence of component tasks, including the task of managing adjuvant therapy which itself decomposes into a decision, a cyclical therapy administration task, and so on.

The graphical notation provides an intermediate representation between an informal description of the protocol or guideline (such as a conventional protocol document) and a machine executable knowledge base. The notation provides a succinct and natural summary which can be understood by medical specialists, and also helps to guide the detailed specification of the tasks by a programmer. In Step 2 of the PROforma method, the detailed clinical knowledge and other information required to populate the task network are added.

2.2. Populating task templates

All clinical tasks are different, but can be modelled in terms of classes which have a common, generic structure. All protocols, for example, have eligibility conditions (preconditions) and conditions under which the protocol should be abandoned (abort conditions). Decisions, actions and enquiries can similarly be modelled as generic tasks. PROforma provides a standard attribute-value template, predefined for each class of task, to support the guideline definition process. Templates guide the application designer in formalising the necessary medical knowledge required for each task to be defined within a guideline or protocol.

Each of the five sub-classes of PROforma task has its own distinct attributes which distinguish it from tasks of other types. The specific attributes of decision tasks are shown in table 1. But each sub-class also inherits a number of attributes from the general class "tasks"; all tasks have a clinical objective (goal), for example.

Table 1: Distinguishing attributes of decision tasks

| Attribute | Description |
|-----------------------|--|
| Candidates | The candidate set of decision options (hypotheses, actions). |
| Arguments | Rules defining “pros and cons” of different candidates. |
| Commitment rules | Rules based on arguments for committing to a candidate i.e. taking a decision. |
| Sources | Any information sources required. |
| Mandatory information | Information essential for commitment of decision. |

From table 1 we can see that all decisions (whether diagnostic decisions, therapy decisions, prescribing decisions, referral decisions etc.) must have a set of options (candidates), information sources which are relevant to choosing between the options, rules of argument to establish preferences among the options, and commitment rules. There are two decisions in Figure 2: one to select the type of surgery (treatment type), and one to select the type of adjuvant therapy (hormonal or cmf). Clearly these decisions are different in detail (we are choosing between different things and need to take into account different patient data and/or general medical knowledge in order to make the choice) but all decisions are modelled using the generic template shown in the table.

3. An example PROforma application

A decision support application, based on a guideline published by the British Thoracic Society [5] concerning the management of acute asthma in adults, has been developed in an early version of PROforma (figure 3).

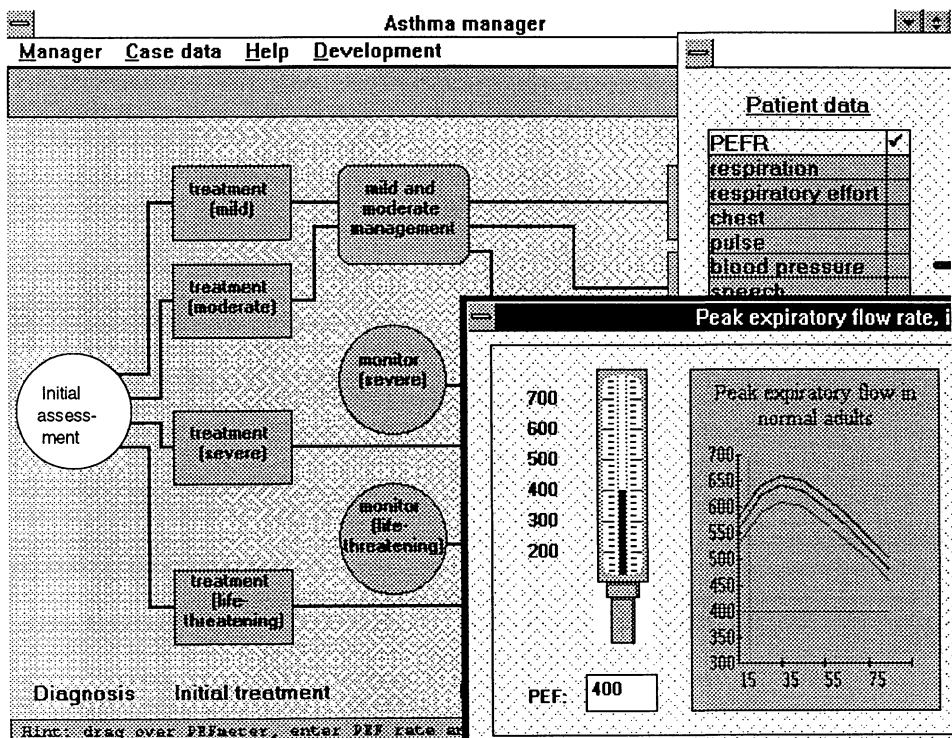


Figure 3 : A user display from a computerised guideline/protocol management system, in this case for the management of acute asthma in adults in the accident and emergency department.

The decisions and actions recommended by the guideline are shown here as a graphical workflow diagram. This can be used passively, to summarise and remind clinical staff of the tasks that need to be carried out according to the protocol, or actively, to assist in the correct and timely execution of clinical tasks: prompting and reminding users about clinical actions which need to be carried out or information that is required, assisting with patient data entry, decision making and so forth.

In the example the first task required by the guideline is an initial patient assessment decision. The two panels on the right are an *aide mémoire* that prompts for patient data that are relevant to the assessment and a data entry screen for the user to enter peak flow and associated information. Depending upon the assessment (mild, moderate, severe or life threatening) care of the patient will proceed along the appropriate branch of the protocol emerging from the decision. In the centre of the picture are two special decisions: these are "watchdogs" that monitor the patient record in order to detect any adverse events or possible hazards that may occur.

Note that the generic notation used in the design of a PROforma protocol is also used here to represent the workflow to the user, augmented with application-specific data-entry forms. Colour coding is also added to indicate various dynamic states of a task, for example to indicate whether the task is currently active, scheduled or completed. Although this notation provides a useful starting point for designing user interfaces, specific applications will often require application-specific displays, such as the peak flow data entry and display screen.

4. Conclusion

PROforma embodies a language and a development method for specifying clinical procedures such as therapeutic guidelines and protocols. It provides a notation for formally specifying a wide range of clinical tasks, and the medical knowledge and patient data associated with clinical decision making. The formalism is based on a generalised model of clinical decision making and protocol management developed over a number of years by the Imperial Cancer Research Fund in collaboration with Institut Bergonié, Bordeaux and other colleagues. Details of parts of the work underlying the development of PROforma can be found in [2,3,4]. A detailed definition of the PROforma language will be published in due course.

Acknowledgements

PROforma is being developed within PROMPT (*Protocols for Medical Procedures and Therapies*), a project supported by the EC 4th Framework Health Telematics programme (1996-8). The work is being undertaken by the ICRF in association with members of the ACTION cluster of projects. Thanks are due to colleagues in the DILEMMA, RED and PROMPT projects (particularly Jean-Louis Renaud-Salis, Ian Herbert, Johan van der Lei and Paul Taylor) for their contributions to the ideas reported in this paper.

References

- [1] M A Musen, S W Tu, A K Das and Y Shahr, A component based architecture for automation of Protocol-Directed Therapy. In: P Barahona, M Stefanelli and J Wyatt (eds) Proc. AIME95, Springer, Berlin, 1995.
- [2] J Fox et al, LEMMA: methods and architectures for logic engineering in medicine in J Noothoven and J P Christensen (eds) *Advances in Medical Informatics*, IOS Press, Amsterdam, 1992.
- [3] R Thomson, Decision support in primary care, oncology and shared care in M F Lairens, M J Ladeira and J P Christensen (eds) *Health in the New Communications Age*, Amsterdam: IOS Press, 1995.
- [4] J Fox, S K Das, D Elsdon and P Hammond, Decision making and planning by autonomous agents: an architecture for safety critical applications In: Proc. Expert Systems 95, Cambridge University Press, 1995.
- [5] Anon, Guidelines in the management of asthma, *Thorax: the journal of the British Thoracic Society*, **48**, 2, Supplement March 1993, pp. S1-S24.