

Internet-based Decision Support for Evidence-based Medicine

Jon Simpson*
Peat Marwick
McLintock
Edinburgh, Scotland

John Kingston**
AIAI,
University of Edinburgh
Edinburgh, Scotland

Neil Molony
Royal Infirmary
of Edinburgh
Edinburgh, Scotland

AIAI-TR-230
December 1998

Published in Applications and Innovations in Expert Systems VI,
proceedings of Expert Systems '98, the annual conference of the British
Computer Society's Specialist Group on Expert Systems

Peterhouse College, Cambridge, UK, 14-16 December 1998

Also in the *Knowledge Based Systems Journal*, 12 (5-6), Elsevier Science,
1999, pp. 247-255.

Artificial Intelligence Applications Institute
University of Edinburgh
80 South Bridge
Edinburgh EH1 1HN
United Kingdom
Tel: +44 (131) 650 2732
Fax: +44 (131) 650 6513
email: J.Kingston@ed.ac.uk
<http://www.aiai.ed.ac.uk/>

* This work was carried out while Mr. Simpson was a student in the Department of Artificial Intelligence, University of Edinburgh.

** Author to whom all correspondence should be addressed. E-mail
J.Kingston@ed.ac.uk.

Abstract

The Protocol Assistant is a knowledge-based system, developed by the Department of Artificial Intelligence and AIAI at the University of Edinburgh, which advises on the treatment of parotid tumours. It has been developed to support both adherence to a clinical protocol based on the latest evidence and the use of clinical judgment where the evidence is weak or inconsistent. It was developed using a knowledge modelling technique named *PROforma*, which is specifically designed for representing best practice guidelines; the *PROforma* models were used as the basis for a user interface, which was implemented in HTML. A set of rules were developed in JESS (the Java Expert System Shell) which were capable of “running” the protocol; a simple method of reasoning with certainties, based on the “goodness” of each relevant item of published evidence, was used to recommend which path to follow at choice points. However, the user is also supplied with access to the abstracts of all relevant published papers, using the hypertext facilities of HTML. The Protocol Assistant can thus be used either as a “wizard” which guides users through the decision making process, or as a “hypertext manual” which leads them to the information relevant to the decision they are making. This dual-role capability is crucial for the acceptance of KBS in the real world.

1 Introduction: Artificial Intelligence in Medicine

Artificial Intelligence in Medicine was a field born during the 1970's amidst the euphoria surrounding the promises of artificial intelligence. At this time there was also an explosion in medical knowledge which was forcing health care professionals to become increasingly specialised. Because of this medicine seemed a logical field to apply the knowledge based techniques that had been developed during the sixties for solving game playing, pattern recognition, and language understanding problems. Despite a number of early successes the dreams of the first ambitious researchers still remain little more than dreams and very few AI systems are actually in routine use in the medical world. This is not due to the technology failing; it has more to do with the poor integration of systems into the clinical working environment and the unwise marketing of expert systems as replacing their human counterparts.

Medical practice is, however, changing due to developments in clinical research and the recent enthusiasm for what has become known as "evidence-based" medicine. Back in 1960 randomised controlled trials were extremely rare and yet now in the nineties it is accepted that practically no drugs are allowed to enter clinical practice without having been proved by a clinical trial. The swing towards basing clinical practice on the best evidence from clinical trials has been considerable and is evident in the sheer number of articles instructing clinicians on how to access, evaluate and interpret medical literature.

Yet evidence-based medicine is not without its problems. The major difficulties occur when published evidence is insufficient to use as a basis for clinical practice, when clinicians are unaware of the most recently published evidence, or (as often happens) different clinical studies produce inconsistent results. The doctor's dilemma can be summed up in the quotation below:

"Those who have been in the profession of medicine, and especially surgery, for any length of time, know that basing every action on previously published *proof* is virtually impossible. Yet to speak against evidence-based medicine is akin to saying that the king has no clothes." [1 - italics added]

A solution is needed which finds a way of using all the evidence currently available to full effect, but also allows clinical judgment and experience to decide on the best practice when there is no clear evidence or when there is conflicting evidence. One of the most promising solutions to this problem is Protocol Assisted Care in which clinical protocols which detail the best-justified procedures for given clinical situations, are prepared by senior clinicians or public health organisations; the advantages of protocol assisted care are listed in [2] and [1], and some of these protocols are sufficiently well respected to be close to mandatory (e.g. the publications of the Scottish Intercollegiate Guidelines Network - see [3]).

Computerised support for protocol-based care has been made available in the form of Internet-based publication [4], Internet-based libraries of protocols and abstracts of published clinical trials (see the website of the Cochrane collaboration [5]), and AI-based systems to provide decision support in following protocols (such as ONCOCIN [6] and EON [7]). However, none of these systems support clinicians in both following best practice protocols *and* in using clinical judgment; publications in text form lack any automated decision support, while ONCOCIN and EON follow protocols in a deterministic fashion without providing access to the supporting evidence for decisions.

The Protocol Assistant is a knowledge-based system which has been by the Department of Artificial Intelligence and AIAI at the University of Edinburgh [8] to support both adherence to a protocol based on the latest evidence and the use of clinical judgment where the evidence is weak or inconsistent. It does this by representing the protocol using a simple yet expressive graphical notation; by providing a rule-based component which “runs” the protocol and asks the user for the necessary information; and by providing hypertext links from the protocol to both the abstracts and the full text of all published clinical trials relating to each decision point. The purpose of this paper is to describe the techniques which were used to implement the system, and was also implemented in a format which would be acceptable to clinical users.

2 Representing clinical protocols

The Protocol Assistant was developed based on knowledge from an experienced ear, nose & throat surgeon, and a terse draft of a text-based clinical protocol which he was developing. The protocol dealt with the diagnosis and treatment of parotid swellings (subcutaneous lumps which appear in the neck, below the ends of the jaw); this is an important aspect of otolaryngological work, since parotid swellings may be malignant, and so swift and accurate diagnosis is important.

2.1 Knowledge acquisition and modelling using *Proforma*

The first decision to be made when developing the Protocol Assistant was to decide how protocols would be represented in the system. Drawing on previous experience of knowledge modelling and analysis using techniques such as CommonKADS [9] and IDEF3 [10], we decided to represent protocols using a node-and-arc based knowledge modelling technique. Modelling knowledge provides an intermediate representation between an expert’s knowledge and the final implemented system; if the modelling technique is good, the models should also be comprehensible to domain specialists, and can therefore be used to support further knowledge acquisition. The models resulting from such a technique resemble flow charts at first sight; however, they differ in some important respects. The models are hierarchically structured, so a single node in the top level model may be represented by one or more detailed sub-models; each node in a model represents a different type of knowledge

(for example, IDEF3 differentiates activities, objects, and AND/OR junctions); and the nodes in a model may have descriptive attributes attached (for example, CommonKADS recommends that nodes which represent tasks or activities should have attributes describing their goal, their inputs and outputs, and the task specification).

The knowledge modelling technique which was chosen for this project was *PROforma*, developed by the Imperial Cancer Research Fund [11]. The *PROforma* language was developed specifically for the task of representing best practice guidelines. It assumes that three types of knowledge are required by a clinician when making patient management decisions. These are :

- General Medical Knowledge
- Specific Patient Knowledge
- Knowledge Of Best-Practice Procedures.

Knowledge representation in *PROforma* uses an ontology of four basic activities. These are shown in the diagram below :

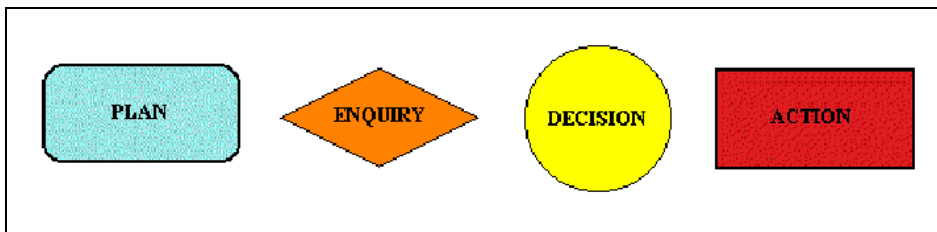


Figure 1: The ontology of activities within PROforma

These activities are defined in the following way:

- A *Plan* is a sequence of sub-tasks, or components, which need to be carried out to achieve a clinical objective, such as a therapeutic objective. Plan components are usually ordered, to reflect temporal, logical, resource or other constraints.
- 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 query on a local patient record, or a remote laboratory database).

- A *Decision* task occurs at any point in a guideline or protocol at which some sort of choice has to be made, such as a diagnostic, therapeutic or investigative choice.
- An *Action* is a procedure which is to be enacted outside the computer system, typically by clinical staff, such as the administration of an injection.

The *PROforma* ontology specifies required attributes for each data type; these attributes proved helpful in specifying the knowledge which needed to be acquired in order to complete the model. It also permits Plans to be decomposed into lower-level models, thus allowing the total number of activities in the whole protocol (about 30 altogether) to be subdivided logically between 8 diagrams. This also simplified the representation of multiple paths to the same conclusion, since each path could be represented in a separate diagram, and then linked to another diagram representing the shared conclusion and its consequences.

Once initial knowledge acquisition had been performed, *PROforma* diagrams were created using Hardy [12], a meta-CASE tool for creating node-and-arc diagrams of various types with additional hypertext facilities for linking between text, individual nodes, and whole diagrams. These diagrams constituted models of the clinical protocol, and the next stage of knowledge acquisition focused on verifying the accuracy of these models. The resulting models were then output by HARDY in a HTML-compatible format: the diagrams are converted into bitmaps and then into GIFs which can be displayed in a frame within a browser, the attributes of each node are stored in a *.htm* file which can be displayed in a separate frame by clicking on the node, and all the hyperlinks are preserved. This HTML representation of the models was used both for displaying the models to the clinical expert in order to facilitate further knowledge acquisition and knowledge refinement, and as a basis for the user interface of the Protocol Assistant. An example of the interface can be seen in Figure 2.

2.2 “Running” a clinical protocol using JESS

Having obtained a representation of a clinical protocol in an HTML-compatible format, the next step in developing the Protocol Assistant was to provide a means of “running” the protocol. Running a protocol implies providing an automated “expert system” which would start at the beginning of the protocol, ask all (and only) the relevant questions, dynamically determine its path through the protocol based on the user’s answers to previous questions, and finally reach a particular end point, thereby suggesting a diagnosis and recommending an approach to treatment. The requirement that the system should dynamically determine its next action based on previous input creates a preference for a production rule-based approach, since production rules take

this approach by default. The chosen tool also needed to be able to obtain input from, and provide output to, an HTML-based user interface.

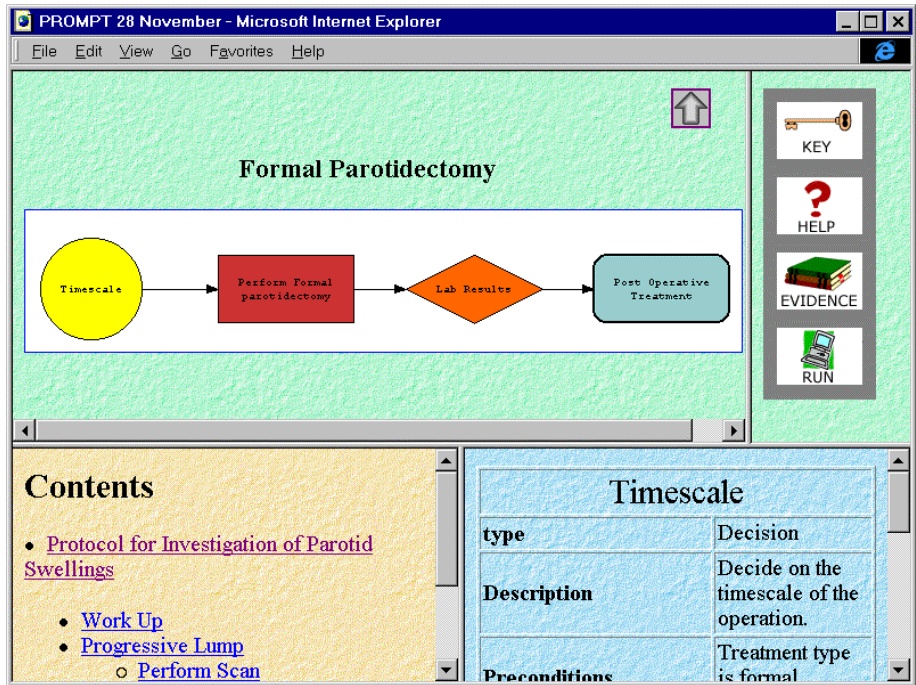


Figure 2: HTML-based representation of a clinical protocol

The chosen tool was JESS (the Java Expert System Shell), a “clone” of CLIPS which was written in Java [13]. JESS is described as “essentially an interpreter for a rule language borrowed from CLIPS”; it therefore supports the development and execution of forward-chaining rules, which are compiled using a RETE algorithm. The major advantage of JESS for this project is that the rules can interact with Java code, thus fulfilling the requirement to be able to work with an HTML-based user interface. Details of the design and implementation of this “expert system” module are given later in the paper.

2.3 Representing and reasoning with clinical uncertainty

An innovative feature of the Protocol Assistant is its ability to represent clinical uncertainty. One of the main motivating factors for this project was to be able to represent protocols for which published clinical evidence was scarce or inconsistent -

and yet hardly any other software for automating protocols supports this feature.* Uncertainty may arise at any Decision node in the protocol, and the Protocol Assistant is capable of representing evidence both for and against particular courses of action.

The assessments of clinical evidence are based on a ranking of the “goodness” of published evidence which was presented in [1]. Randomised control trials provide the “best” evidence, while unsupported opinions from respected authorities are considered the “weakest” evidence. The full ranking is as follows:

1. (a) Evidence obtained from meta-analysis of randomised control trials
(b) Evidence obtained from at least one randomised control trial
2. (a) Evidence obtained from at least one well-designed controlled study without randomisation
(b) Evidence obtained from at least one other type of well-designed quasi-experimental study
3. Evidence obtained from well-designed non-experimental descriptive studies such as comparative studies, correlation studies and case-controlled studies
4. Evidence obtained from expert committee reports or opinions and/or clinical experience of respected authorities

In order for the Protocol Assistant to reason about these types of evidence, a relative scoring system had to be devised: for example, do five expert committee reports outweigh a single randomised control trial? Since the “strength” of each type of evidence actually represents a level of certainty that the evidence is accurate, recognised AI methods of reasoning with measures of certainty were considered. Bayesian probability and Dempster-Shafer theory have the advantage of a strong theoretical basis, but were rejected because of the “loss of comprehensibility” which can arise when these theories are applied to real world situations (i.e. the propagated numerical certainty values can be hard for users to comprehend). MYCIN-style certainty factors provide a simple method for handling uncertainty, but they lack the theoretical weight of the other two numerical approaches. Cohen’s theory of endorsements [14] is a particularly intuitive way of handling uncertainty, but it doesn’t have any second order measure of uncertainty which makes combining evidence difficult. Fox’s logic of argumentation [15] has a great deal of potential, but it is currently not easy to interpret how to use the method in a practical situation since most of the published work is theoretical.

It was decided that an approach based on endorsements would be used, utilising some of the ideas proposed by the Logic of Argumentation. To address the weakness of

* The only known package which claims to offer support for this feature is PROMPT, which is based on *PROforma* and was developed by the Imperial Cancer Research Fund [11]. PROMPT was not used for this project for a number of reasons, including the desire for Internet-based delivery.

endorsements, the ranking of evidence types described above has been used as a second order measure of uncertainty. All evidence relating to a particular proposition can then be combined to decide which advice the evidence indicates the system should give. The varying weights applied to different combinations of evidence are based on the results of a knowledge acquisition session with Dr. Molony; this session used a set of contrived “cases” to discover the relationships between the different quality of evidence. Once suitable weightings for each type of evidence have been established, it is then possible to combine the different strengths of evidence and recommend actions based on the results.

The “algorithm” for combining weights was simple addition and subtraction. This was used because that it ought to be adequate if the weightings were calculated well; the number of items of evidence in each calculation was small (often less than 10, sometimes less than 5), reducing the utility of more complex calculations; and the project was viewed as an empirical test of a very simple approach, on the basis that it’s often wise to use the simplest approach which works.

However, one of the major benefits of the Protocol Assistant for the user is that it does **not** enforce its choice of the “best” decision to take at each decision point. Instead, it suggests the best decision based on its certainty calculations, and then offers all the evidence to the users so that they can make up their own mind. By clicking on a Decision node, the users can view a list of relevant published articles (including conflicting evidence, if any), and can use the hypertext features of HTML to read the abstract of that article, or even to read the full paper. This feature allows a user to employ the Protocol Assistant either as a “wizard” which guides them through the decision making process, or as a “hypertext manual” which leads them to the information relevant to the decision they are making. This dual-role capability is crucial for the acceptance of KBS in the real world, since different users have very different requirements of fielded systems, and can quickly become irritated if the system does not meet their requirements.

An example in which the Protocol Assistant displays published evidence can be seen in Figure 3.

3 Design and implementation of the Protocol Assistant

3.1 System Design

The design of the overall system was initiated by preparing a *use case diagram*, which describes the desired behaviour of the system from the users’ point of view. Use case diagrams consist of two elements; actors shown as a stick figure and use cases shown as a named oval. An actor is a “human user of the system in a particular role” or “an external system which in some way interacts with the system.” A use

case is defined as “a coherent work unit of the system which has value for an actor” [16].



Figure 3: A published paper which gives evidence for a particular decision is presented by the Protocol Assistant

From the use case analysis (see Figure 4) we can see that a typical doctor using the system would want to be able to enter patient data, receive advice on actions, view clinical protocols and view the evidence supporting the protocols. We can also see that the experts are responsible for defining the clinical protocols and that the World Wide Web supports a system with a user interface that can collect the evidence for a protocol. The final actor in the diagram is the patient who provides case specific details which would be entered and used to provide advice.

The main focus of this project - to generate advice on which action should be performed - is indicated by a note on the diagram. The storage of data to allow stop-start use was identified as a possible future requirement but due to complications with the legal aspects of storing patient data it was decided to leave this as a future extension.

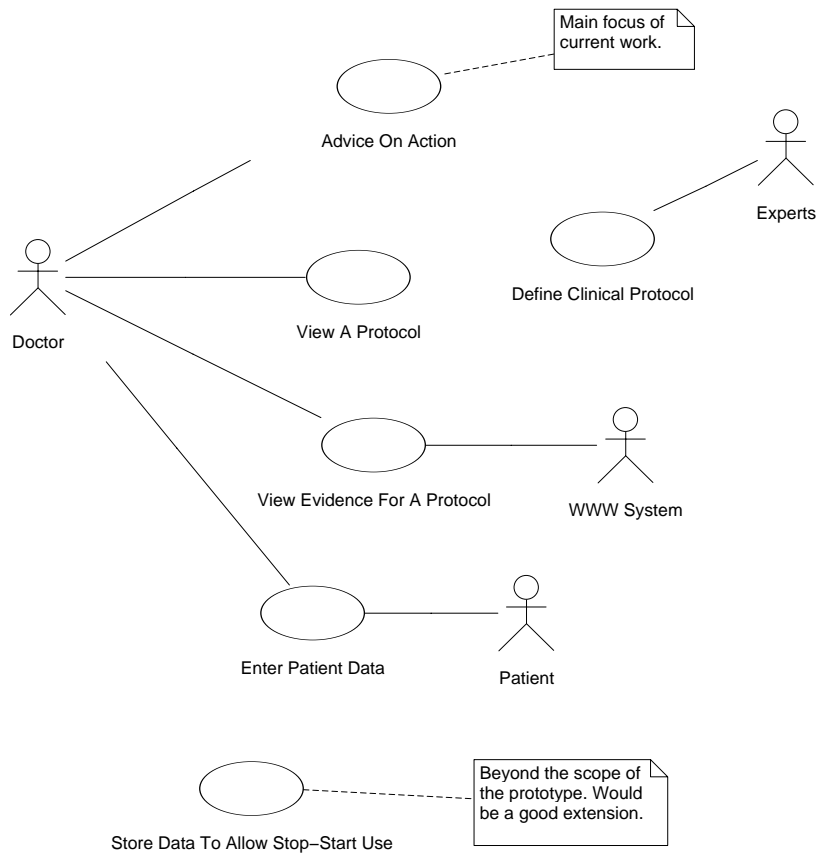


Figure 4: Use Case Analysis

The UML component diagram shown in Figure 5 represents the run-time dependencies of the five components of the system. Starting at the rule level the CLIPS rules are interpreted into Java using the Java Expert System Shell (JESS). The user interface component provides an interface to the rules interpreted by JESS and allows the user to interact with the system from a Web browser. The Web browser interprets the HTML pages and is responsible for display the pages in the appropriate frames. The Web browser also interacts with the Java applet and

receives requests from the applet to display particular HTML pages as the applet runs.

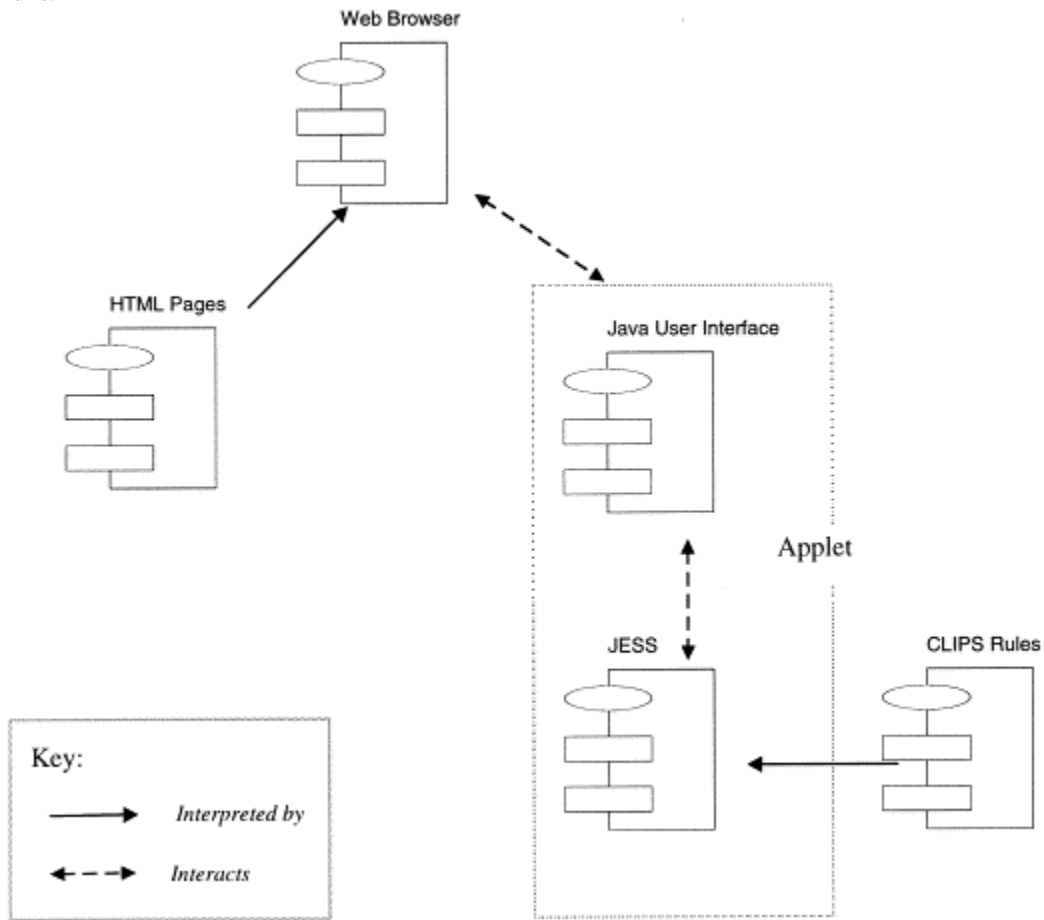


Figure 5: Component Diagram

3.2 User interface design

One of the challenges when developing a medical expert system is getting the medical profession to accept and use the system. Because of this difficulty, considerable thought was given to the user interface so that the system would be as intuitive as possible for clinicians to use. To help maintain consistency throughout the interface a template was designed for the HTML pages, containing four frames. The top left-hand frame is used to show the protocol diagrams that were produced during the knowledge acquisition stage of the project. The top right-hand frame contains four control buttons that can be pressed at any time to take the user to a particular part of the system. The bottom left-hand frame is initially used to display a table of

contents for the protocol and allow users to jump to examine any point in the protocol. Once the user has clicked on the “Run” button, the bottom left-hand frame is used as a notebook to record the information entered by the user so that they can check back over what they entered as the protocol proceeds. This recording is intended to capture the notes the clinician would normally take as a case is managed. The final frame is used as an information frame to display the online help menu and other information associated with particular steps in the protocol.

3.3 Implementation

For the prototype system, it was decided to implement an “all or nothing” approach to running the protocol; that is, the user must run the protocol from the beginning every time, rather than clicking on a node part way through the protocol and initiating the run from there. This avoided a number of problems, not least of which was deciding how to gather data which would normally have been obtained from the user earlier in the protocol. A set of CLIPS rules were therefore prepared, tested in CLIPS, and ported into JESS; a Java applet was then designed (using Java Development Kit 1.1) which could invoke JESS and also display input forms to a user when JESS requested data. The interaction between rules in JESS and the Java applet was less straightforward than expected; not only is JESS, as its web page says, “work in progress”, but at the time when the Protocol Assistant was being developed, there were very few examples of how JESS could be used. The only working example that was available was the classic “monkey and bananas” program running as an applet, but this system ran from beginning to end without accepting any intermediate user input. This meant that the user interface classes and the method of linking them to JESS had to be developed from scratch without any examples to base them on.

With the assistance of an active JESS mailing list and some additional Java code, the rules were successfully linked to the user interface, so that activating certain rules not only performs reasoning, but also changes the display in the top left frame to reflect the stage of the protocol which is being carried out. The applet can be triggered by clicking on the “Run” button in the top right frame of the user interface; an example of the running system can be seen in Figure 6.

4 Evaluation and future work

When evaluating a system that is intended for real world use, one of the major criteria for evaluation must be how well the system meets the requirements of the users. In this section we will look at six user requirements and consider how well the Protocol Assistant meets them.

In a study of physicians’ attitudes towards clinical consultation systems [17] the following six design features were identified as most important for consultation systems:

1. they should be able to explain their diagnostic and treatment decisions to physician users;
2. they should be portable and flexible so that the clinician can access them at any time and place;

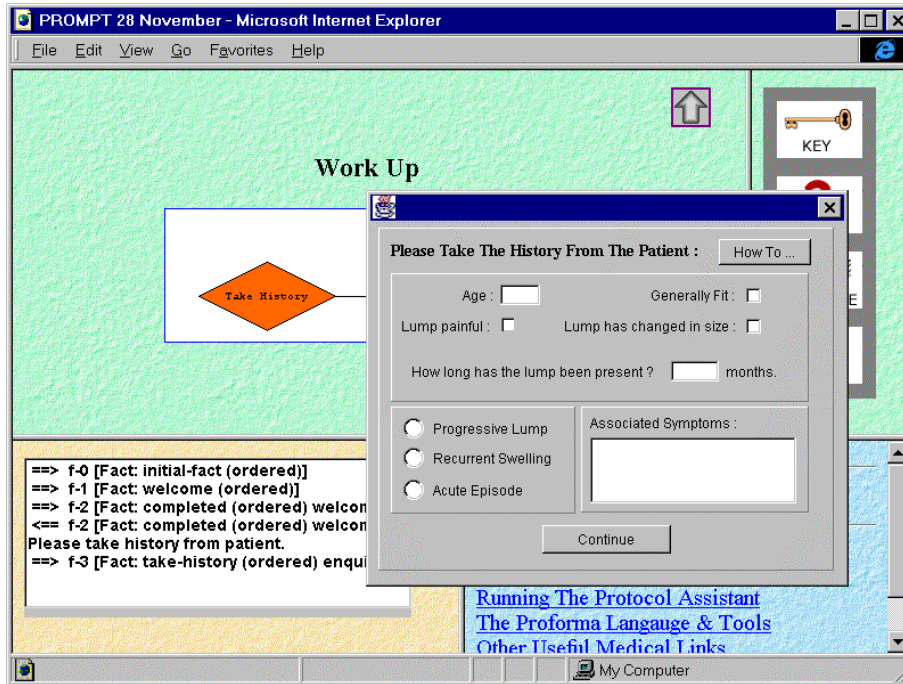


Figure 6: An input form appears during a “run” of the Protocol Assistant

3. they should display an understanding of their own medical knowledge;
4. they should improve the cost-efficiency of tests and therapies;
5. they should automatically learn new information when interacting with medical experts;
6. they should display common sense.

The first requirement is met within the Protocol Assistant by allowing the user access to the clinical protocol diagrams and the online evidence supporting the decisions recommended. By examining these, clinicians can easily find out why the system is recommending a particular decision and make an informed choice about following the recommendation.

The second user requirement of portability is also met by the design of the Protocol Assistant since all that is required to run the system is access to the World Wide

Web. The system can be run using any operating system and can also easily be run locally on a portable laptop computer, as was used for demonstrating the system to the ENT department at the Royal Infirmary of Edinburgh.

The remaining requirements are more subjective and hence less easy to evaluate although it is possible to offer some opinions on how well these requirements have been met. From a purely A.I perspective the system clearly has no conscious understanding of the recommendations it gives since it is simply following a path through a rule base. However, when using the system it can be observed that medics quickly become convinced that the system does have some inherent understanding of the reasons behind the advice it is giving and have been known to verbalise such beliefs by saying things like "that must be because it thinks X". So whilst in reality the system has no understanding of the advice it is giving it appears to convince some users that it does; it must therefore be displaying medical knowledge in a convincing manner.

To provide conclusive evidence that a system could improve cost efficiency would require a long-term study comparing the results before and after introducing the system. Since no such study has yet been carried out, it is difficult to justify suggestions that the system meets this criterion. However, since the system recommends the best justified practice it should reduce the number of non-essential tests that are performed and so may help to improve cost efficiency that way.

The next requirement, that the system should learn from interaction with expert users, is not met by the system. However, given that the system is based on the best knowledge supported by the current evidence, changing the knowledge base through interaction with users would seem like a bad idea. The system should of course be kept up to date with any new evidence that comes to light and by doing this it should be possible ensure that the advice given is the best supported at any given time. A beneficial avenue of research would be to investigate the feasibility of establishing links to relevant publications on remote sites on the Internet (particularly the Cochrane Collaboration's collection of material), thus providing a much wider collection of evidence for users to study.

The final user requirement identified is to show common sense. Again this is a fairly vague notion and is not something that was specifically addressed as part of this project. However, some of the decisions that are recommended by the system are based in some degree on common sense such as recommending against performing operations on very elderly patients because the shock of undergoing an operation may be more dangerous than the tumour.

Further evaluation was carried out by asking potential users with a range of expertise (ENT surgeons, junior doctors and medical students) to answer questions about the need for the system; the expertise level of the system; the usability of the system; the likely impact on patient management and well-being; and the cost-effectiveness of the system [18]. The results show high opinions of the usability, expertise level, and

desirability of the system; cost effectiveness was more difficult to estimate. Perhaps the most compelling argument for the system, however, is the need for better availability and application of clinical protocols; at present,

- there are still very few protocols that have been published;
- protocols that have been published don't use the same way of describing procedures;
- clinical protocols are likely to change as new evidence comes to light;
- many of the advantages are nullified if an out of date protocol is used;
- volumes of literature are huge with approx. 360,000 articles published in medical journals every year.

For these reasons, the Protocol Assistant is evaluated as supplying sufficient benefits to be worthy of further commercial design and development.

Acknowledgements

We would like to acknowledge the support of John Fox of the Imperial Cancer Research Fund in providing us with documents describing PROforma and ICRF's PROMPT tool [11], and the support of members of the Department of Otolaryngology at the Royal Infirmary of Edinburgh.

References

1. Maran A.G., Molony N.C., Armstrong M.W.J., and Ah-See K. Is there an evidence base for the practice of ENT surgery? *Clinical Otolaryngology* 1996; 22:152-157.
2. Coiera E., Baud R., L. Console, J. Cruz, J. Durinck, P. Frutiger, P. Hucklenbroich, A. Rickards, and K. Spitzer. The Role of Knowledge Based Systems in Clinical Practice. In *Knowledge and Decisions in Health Telematics - The Next Decade*, P. Barahona and J.P. Christensen (eds), 1994:199-203.
3. Scottish Intercollegiate Guidelines Network, *Helicobacter Pylori: Eradication Therapy in Dyspeptic Disease*. Scottish Intercollegiate Guidelines Network, Royal College of Physicians, Edinburgh, August 1996
4. Detmer W.M., and Shortliffe E.H., *Using the Internet to Improve Knowledge Diffusion in Medicine*. Communications of the ACM, 1997.
5. The Cochrane Collaboration, *Abstracts of Cochrane Reviews*, <http://www.hcn.net.au/cochrane/abstracts/intro.htm>
6. Musen M.A. and Johnson P.D. Development of a guideline authoring tool with PROTEGE II, based on the DILEMMA Generic Protocol and Guideline Model. Section on Medical Informatics, Stanford University School of Medicine, 1997.
7. Shortliffe E., *Studies to Evaluate the ONCOCIN System*. Stanford Heuristic Programming Project Report HPP 84-22., Stanford University, 1984.

8. Simpson J., An Expert System for "Best Practice" in Medicine, Project Report, Department of Artificial Intelligence, University of Edinburgh, June 1998.
9. J.K.C. Kingston, Re-engineering IMPRESS and X-MATE into CommonKADS, in Research and Development in Expert Systems X, Proceedings of Expert Systems 93, St. John's College, Cambridge, 13-15 December 1993. Cambridge University Press, 1993.
10. John K.C. Kingston, Anna Griffith and Terri J. Lydiard, Multi-perspective modelling of Air Campaign Planning. In Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI '97), Nagoya, Japan, 23-29 August 1997. AAAI Press, 1997.
11. Fox J., N. Johns and A. Rahmanzadeh, Protocols for Medical Procedures and Therapies : A Provisional Description of the PROforma Language and Tools. In Proceedings of AIME '97. Springer Berlin, 1997.
12. Smart, J. HARDY. In Airing, the magazine of AIAI, Edinburgh, April 1993:3-7. See also Hardy Information, <http://www.aiai.ed.ac.uk/~hardy/hardy/hardy.html>
13. E. Friedman-Hill, JESS home page and JESS manual, <http://www.hertzberg.ca.sandia.gov/jess>
14. Cohen, P.R. Heuristic Reasoning about Uncertainty: An Artificial Intelligence Approach. Pitmans Boston, 1985.
15. Krause P., Ambler S., Elvang-Goransson M. & Fox J. A logic of argumentation for reasoning under uncertainty. Computational Intelligence 1995; 11 (1).
16. Stevens P. Software Engineering with Objects and Components. Computer Science 4 Lecture Notes, University of Edinburgh, 1997.
17. Teach R., and Shortliffe E. An analysis of physician attitudes regarding computer-based clinical consultation systems. Computers and Biomedical Research, 1981;14:542-558.
18. Shortliffe E., and Davis R. Some considerations for the implementation of knowledge-based expert systems. SIGART Newsletter 1975; 55:9-12.