Introduction

The development, implementation and maintenance of computer-executable clinical guidelines delivered within point-of-care decision-support systems is a multifaceted, resource-intensive process. From a knowledge perspective, these processes may be thought of as: knowledge acquisition, knowledge encoding, guideline execution (making knowledge available to clinicians) and knowledge base maintenance. In practice, the usual pattern is that a guideline development group will produce a clinical guideline and this is used as the source for encoding into a computer-interpretable guideline. This is then executable within a decision-support system available to the clinician. Guideline maintenance is necessary to incorporate new evidence into the knowledge base.

This model of production generates a series of problems and challenges at each step in the process. For example, traditional guideline development—which may be only the first step in this process—is in itself very complex. So is the encoding process—Tierney et al. describe the difficulty of encoding just one guideline. Guideline execution requires the integration of the guideline knowledge with the clinical computer system via an ‘execution engine’. Guideline maintenance consists of identification of changes in evidence for recommendations, their incorporation into the knowledge base, external validation and release of guideline to the end-user.

There are some fundamental requirements in order to deploy this approach in clinical practice. The goal of the original guideline development is ultimately to influence patient outcomes through changes in clinical practice. To be realised, the original recommendations constructed by the guideline author need to be faithfully maintained throughout the knowledge transfer and relayed to the end-user. In addition, the process must be scalable to enable wide-scale roll-out. A major contributor to scalability is knowledge sharing and/or re-use. This is often hampered because each part of the process requires different sets of skills and so consequently is carried out in disparate organisations or parts of an organisation. Duplication of effort, both within and between organisations, is often an inevitable consequence. This has led to many initiatives to address knowledge sharing and system interoperability,
as demonstrated by Greenes et al.’s work on GLIF (Guideline Interchange Format).4 The ever-increasing number of formalisms that are developed to represent knowledge also contribute to the problem.5

PRODIGY is a major national programme of work to create, deliver and implement guideline-based point-of-care decision support to primary care physicians.6 It is being developed at the Sowerby Centre for Health Informatics at Newcastle (SCHIN). Within the centre, all aspects of the guideline production process are controlled, bringing together knowledge authors, technical authors, software programmers, evaluators and disseminators to develop the knowledge base, the software requirements specification (SRS) and an execution engine. The knowledge authors, working within an annual budget of £1.3 million, are a large multidisciplinary team of clinicians, pharmacists, technical authors and researchers. The product that is currently available in United Kingdom (UK) general practices contains 131 individual guidance topics, covering over 350 individual scenarios (distinct types of presentation of patient or condition).7 It is implemented on ten different systems covering over 90% of the market. The current research phase (Phase 3) of PRODIGY is developing methods for improving the management of chronic disease in primary care.

This paper describes the evolution of the PRODIGY knowledge base. It addresses some of the challenges of guideline encoding, discusses the requirements for representing knowledge in two different PRODIGY models, and proposes future research and development activities in this area.

The PRODIGY knowledge base and guidance model

The PRODIGY knowledge base currently spans two models – Release 1 and Phase 3 (Release 2).

Release 1

The structure of the Release 1 model can be seen in Figure 1.8 Each level of the model contains further information. The ‘guidance’ level – guidance are individual topics such as asthma, sore throat, dyspepsia – contains a series of text fields such as ‘Background information’ or ‘Management issues’, each of which contains information relevant to the entire guideline. At the level of the scenario similarly – a scenario is an easily recognisable patient state for a particular diagnosis, for instance, ‘angina on triple therapy’, or ‘hypertensive on non-pharmacological treatment’ – there are supportive texts, this time relating only to the scenario. Each scenario contains ‘therapy groups’ (these are broad groupings of therapies, usually drugs, grouped by drug class or by some other common property of the prescriptions contained within, for example, dose frequency). Each therapy group contains prescriptions. Additionally, patient information leaflets (PILs) are available within each scenario.

PRODIGY integrates with the electronic patient record (EPR) so that patient data items such as age or sex will filter out certain guidelines and/or scenarios so that the information that is displayed to the user is specific to the patient. Each guideline is also coded with a set of clinical codes (4-byte or Version 2 Read code or Clinical Terms Version 3) to allow appropriate guidance to be ‘triggered’ when a clinical term is recorded during the doctor/patient interaction. Once the guidance has been triggered, the clinician has options to access supporting texts (as described) or to proceed to select therapy options and/or PILs. The encounter ends with the issue of a prescription, the issue of a PIL, or simply with advice or reassurance in the usual manner.

Release 2

During the next design phase (Phase 3), the structure of PRODIGY Release 1 was analysed for inclusion in the new model. The primary reason for developing Release 2 was to support the management of chronic disease in primary care more effectively. To achieve this, we incorporate the ability of the system to ‘remember’ its previous status throughout a series of consultations over time. This means that the clinician does not have to manually position their patient in the guideline each time the patient attends. The system
can predict the future position of the patient in the guideline based on actions that are taken at each successive encounter. Other key features are: structuring the knowledge base so that it interacts more closely with the EPR (for example, being cognisant of co-morbidities and co-prescribing) and to enable ‘ordinary’ clinicians to construct the guideline.

PRODIGY Release 2 is built as an ontology using Protégé, a frame-based knowledge-authoring environment. A set of classes is constructed representing the knowledge to be acquired. A diagramming facility allows the author to visualise spatial relationships between guideline elements and to visualise the guideline globally. The structure of Release 2 is again constructed around scenarios. The design of scenarios has to take into account the fact that the criteria for the scenario being true must be computable (for example, the presence of a drug or other data item). This precludes the possibility of having scenarios built on concepts that cannot (will not) be represented in the patient record. This was not a requirement in Release 1. Scenarios are linked together by action steps (see Figure 2).

The action that is performed determines the next scenario. ‘Consultation’ actions (history taking, investigations, referral, and PILLs) are offered independently of the scenario. Supportive texts are provided in the form of ‘quick-helps’, background reference documents, and as ‘context-sensitive help’ from individual actions. A state transition diagram (see Figure 3) – which maps out the guideline – is constructed for each condition. This also allows access to each element that contains further levels of detail.

We have also created a PRODIGY drug dictionary in which each product is identified with a clinical drug term (Read code) and mapped to two proprietary drug dictionaries to enable implementation in UK primary care (Release 1 only uses Read codes).
Knowledge acquisition and encoding

During Phase 3, clinical and technical authors collaboratively developed three chronic disease guidelines: angina, asthma and hypertension. This consisted of, firstly, acquiring the knowledge and, secondly, encoding the clinical content of the guideline into a computer-interpretable form so that it can be delivered as decision support within the user’s clinical computer system.

Knowledge acquisition is the identification, collection and appraisal of source evidence. In Phase 3, we used existing peer-reviewed, paper-based guidelines. Encoding consisted of analysing the source guideline to identify:

- **scenarios**
- all possible actions, e.g. prescribing, investigations, referral, advice and so on
- **concepts** that the guideline needs to be aware of, e.g. co-morbidities, patient populations, risk factors and so on; these form the basis of criteria that affect the presence or preference of possible actions.

Each element is then represented within Protégé and populated with information, such as logical criteria to determine preference, help texts and so on. As the encoding process proceeded, each guideline was tested using a prototype user interface that facilitated quality assurance. Although the source guidelines are used as the source evidence for the exercise, and are often thought of as being ‘complete’, there were many occasions where we had to disambiguate statements that were made. In addition, we had to supplement the guidelines with other sources of information.

Knowledge re-use

This takes three main forms in the PRODIGY programme.

Re-using knowledge from Release 1 in future models, e.g. Release 2

There are several reasons why it is desirable to migrate knowledge between different guideline models:

- to avoid wastage of resource on re-representation of knowledge
- to develop and demonstrate a sharable methodology
- to minimise work for clinical system suppliers
- to maximise the exposure of content to users of new and legacy systems.

To this end, we analysed the structures of the two models to identify commonality. The working hypothesis was that we could partially automate the conversion of Release 1 content into the Release 2 model. The first step was to identify equivalent elements in the models. Those items that were deemed equivalent were flagged to be automatically transferred to create an instance of the class to which they belonged. Any elements that were not equivalent were analysed to determine whether they could populate any element in the new model. In fact no element in Release 1 was redundant. Examples of the process of determining equivalence are shown in Box 1.

**Box 1 Equivalent elements in PRODIGY**

<table>
<thead>
<tr>
<th>Guideline level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Background texts (R1) → Reference document (R2)</td>
</tr>
<tr>
<td>Sex and age range (R1) → Eligibility criteria (R2)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Scenario level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Which therapy text (R1) → Scenario Quick help (R2)</td>
</tr>
<tr>
<td>Sex and age range (R1) → Precondition (R2)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Action level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Therapy group (R1) → Action step (R2)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Prescription level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Therapy detail (R1) → Prescribable item (R2)</td>
</tr>
</tbody>
</table>

R1 – Release 1, R2 – Release 2
Sharing knowledge across and within guidelines

Knowledge re-use within and across guidelines is being explored within the current research activity. In Release 1 there was very little re-use. In the new model, the design allows for re-use of elements such as actions or sub-guidelines. For example, in the angina guideline, wherever it is appropriate to offer the substitution of a drug, a sub-guideline is used – this only had to be authored once. We are also working on developing common sharable ‘parts-of-guidelines’ or ‘guide-lets’ that could be deployed in any guideline that required that functionality, such as angina, hyperlipidaemia and obesity, which may all require the availability of a lipid screening ‘guide-let’.

Development of related projects that will link to PRODIGY guidelines, e.g. primary care drug dictionary and drug ontology

Currently, drug information is authored for each guideline with only minor re-use. The drug ontology/dictionary will provide significant proportions of drug content for the guidelines and ensure accuracy and consistency.

Discussion

We have described the process of taking clinical evidence within a guideline, encoding it, making it available to clinicians and achieving a degree of knowledge re-use. The likelihood of success can be increased by improving the accuracy of conveying intended meaning, and by achieving scalability.

Conveying intended meaning

This process needs accurately to convey the meaning of the original clinical recommendation. Ideally the ‘encoder’ should not have to interpret the meaning of any terms or concepts that the guideline contains – any disambiguation that is required will require extra resource and may alter the intended meaning. The problem arises out of the need for the encoder to make the concepts computable; for example, the term ‘eating disorder’ may be suggested as a concept in a guideline, but if this is an eligibility criterion for the guideline, then this needs to be interpreted: does it mean all eating disorders or specific ones? This is not just a computability issue; the clinician who uses the guideline would also have to attempt to guess the intended meaning. Therefore, we require guideline developers to be much more explicit in the terms that they use. In PRODIGY we have also internalised the clinical guidance development process, and consequently can be closely involved in getting the clinical authors to be explicit in conveying the meaning of their guidance. There is also a requirement to provide sufficient detail for the encoding process. To this end, work is progressing within the HL7 (Health Level 7) framework to develop architectures for guidelines, one benefit of which will be to specify levels of detail and reduce errors of omission and ambiguity.11

Achieving scalability

The process needs to be scalable. There are several factors that govern scalability; the main ones are: the provision of authoring tools that can be used by ‘ordinary’ clinicians who can be rapidly trained in the skills to encode clinical guidelines into a computer-interpretable format, and the optimisation of the management of knowledge within and across guidelines and guideline representation models to achieve significant degrees of re-use. As part of this we are also exploring the possibility of authoring knowledge content in Release 2 format and ‘backwards populating’ Release 1. This would mean we could continue to implement PRODIGY in two different formats – supporting users of new systems and users of legacy systems – whilst minimising the additional workload in authoring and in supplier implementation.

Acquiring and representing knowledge is a difficult and time-consuming task. It also remains a major cost in developing a guideline-based decision-support system. The proposed developments would go a long way to improve the situation.

ACKNOWLEDGEMENTS

Thanks to the Knowledge Authoring Team and Software Team at the Sowerby Centre for Health Informatics at Newcastle.

REFERENCES

7 Sowerby Centre for Health Informatics at Newcastle, PRODIGY: www.prodigy.nhs.uk

11 Health Level 7: www.hl7.org

ADDRESS FOR CORRESPONDENCE

Richard Hall
Sowerby Centre for Health Informatics at Newcastle University of Newcastle
16/17 Framlington Place
Newcastle upon Tyne NE2 4AB
UK
Tel: +44 (0)191 243 6100
Fax: +44 (0)191 243 6101
Email: richard.hall@ncl.ac.uk

Accepted July 2002