Developing an Evidence Based Medicine Decision Support System Integrated with EPRs Utilizing Standard Data Elements

Jorma Komulainen[1], Ilkka Kunnamo[1], Peter Nyberg[1], Minna Kaila[2], Taina Mäntyranta[3], and Maritta Korhonen[4]

Abstract. We describe an evidence-based medicine decision support service currently under construction. The system will utilise data from various electronic patient records, compatible with the Finnish national EPR standards. The technological architecture is based on SOAP messages over http, and open standard interfaces. Databases for the decision support are targeted for both primary and secondary health care, and include data relevant for doctors, nurses and other health care professionals. To ensure the high quality of the decision support, the databases will be centrally managed, including the evidence-base, functional descriptions, computer-readable scripts, and their metadata.

1 Introduction

A reliable and effective electronic decision support system of clinical use requires both a structured and standardized electronic patient record (EPR) and a reliable scientific evidence base. As part of a national health reform in Finland the core data elements of the EPR have been defined (Table 1). The data are chosen and adjusted to meet the needs of a comprehensive decision support system. A uniform format of the core data – HL7 Clinical Document Architecture (CDA) Release 2 – is mandatory by the end of the year 2007.

Evidence based medicine (EBM) has a long tradition in Finland. The first national Current Care guidelines were published in 1997, and currently altogether 67 guidelines are freely available in the Internet [1]. In addition to that, the Evidence-Based Medicine Guidelines (EBMG) database [2], including about 1000 clinical guidelines developed since 1988, is available for most of the practicing physicians in the country as part of a national health portal [3]. Altogether more than 1000 physicians have been participating in the guideline development. The usage rate of these databases is high among both physicians and nurses.

The implementation of the EBM guidelines is a major undertaking. Recent systematic reviews have shown that automatic electronic decision support systems, especially when available at the point of care, can be effective in supporting professionals in decision-making [4,5].

[1] Finnish Medical Society Duodecim, Helsinki, Finland, [2] University of Tampere, Finohta/Stakes and Pirkanmaa Hospital District, Tampere, Finland, [3] The Centre for Pharmacotherapy Development, Helsinki, Finland, [4] Pohjois-Savo Hospital District, Kuopio, Finland, e-mail: jorma.komulainen@duodecim.fi

2 Definition of the EPR Core Data Elements

A Decision in Principle by the Council of State of Finland on securing the future of health care in 2002 stated that a nationwide electronic patient record would be introduced by the end of 2007. According to this strategy, the common content and structure to be used in every EPR system in all health care organisations in Finland were defined (Table 1).The defined core data elements will be utilised by all EPR vendors and health care organisations (Table 1).

Table 1. The Finnish National EPR Core Data Elements and the code sets used for decision support (available at www.hl7.fi).

Identification data	Patient identification, gender	National ID code, HL-7 codes
	Caregiver identification	OID code
	Care context identification	OID code
Clinical data	Problems and diagnoses	ICD-10, ICPC-2
	Investigations/laboratory tests	National codes
	Procedures	Nordic codes
	Medication	ATC codes
	Health hazards	ICD-10, text
	Physiological measurements	National codes, text
Other data	Treatment plan	National codes
	Medical certificates	National codes

3 Technological Architecture of the Decision Support System

The decision support (DS) system consists of four components (Figure 1).

The first component is a client side component which combines the necessary data, such as the key data from the EPR and context information, and sends it to the DS service. The client side component also shows reminders, prompts and alerts received from the DS service to the user. The client side component can be a part of an EPR system or an independent component.

The second component is a service interface which defines the input and output parameters and operations performed by the DS service. The interface is described as a web service according to the WSDL (Web Services Definition Language) standard. The WSDL interface defines operations of the DS service, SOAP messages for these operations, structure of the messages as XML schemas, protocol (http) and address of the service. Publishing the interface as a WSDL document makes the

interface technology-independent and facilitates considerably the construction of the client side applications.

The third component is a DS database that contains the DS scripts. The scripts are JavaScript functions which are executed by a script interpreter in the DS service. Metadata are defined for the codes and code values used by the scripts. The database is available in a national repository, and it will be automatically downloadable and updatable to locally installed decision support servers. JavaScript was chosen as the script language because of its wide use, simplicity and ease to integrate with user interfaces and interactive forms.

The fourth component is the DS service used by EPR systems through the service interface. The DS service is responsible for the selection logic of the scripts from the DS database and execution of the scripts. The DS service also composes the reminders and alerts produced by the scripts, and sends the SOAP message back to the client side component. The DS database consists of several scripts but only scripts that are relevant for the current patient data are executed. The metadata of the scripts and patient data are used for filtering in the selection logic.

In our implementation we have used Apache Axis as SOAP platform and Apache Tomcat as a web server; the database is MySQL. Application development is based on Java and JavaScripts.

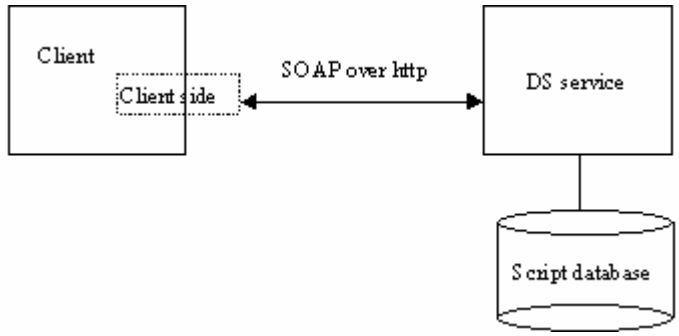

Figure 1. The basic architecture of the decision support system.

4 Object model, script syntax and namespaces

Based on a pilot analysis of 60 script topics, the properties, methods and events of the EPR, the DS service and the clinical guidelines have been described in an object model (OM) (Figure 2). The OM is used for defining the input variables and output messages of the DS scripts. The main object is the Record object, which contains the data output from the EPR to the DS service. The Script object defines the content of the output messages from the DS service to the client EPR. Finally, the Guideline object handles recent guideline changes and updates, which affect the DS Service.

A simplified, easy-to-learn script syntax based on the Visual Basic Script (VBS) language has been developed for the clinicians developing the scripts (Appendix 2). From these "raw" scripts, testing scripts designed for manual input and run by Active Server Pages (ASP) are generated automatically. The final JavaScripts (Appendix 3) are written by information technology experts and are tested extensively. The conversion from the raw script syntax to the Javascript syntax will be partly automated.

Since internationally approved coding classifications are not available for all aspects of the EPR data, the sources of all codes are declared using namespaces. An extensive metathesaurus based on the Unified Medical Language System (UMLS) developed by the National Library of Medicine, USA, enables switching between classifications (including SNOMED CT), which is important in international use.

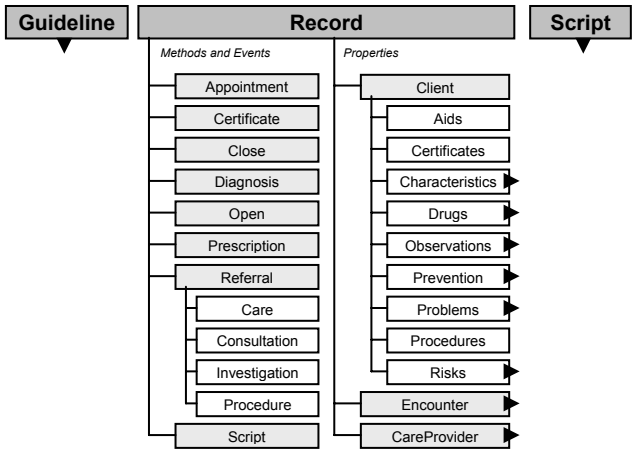

Figure 2. An extract of a summary description of the object model.

5 Functionality of the Decision Support

The decision support is launched with specific triggers initiated from the EPR. These triggers include opening the patient's data in the EPR, prescribing a new drug, proposing a new diagnosis or problem, and ordering a test or a medical procedure.

When a patient's file is opened in the EPR, it sends specified data in CDA R2 format to the decision support service. This basic data includes patient's previous diagnoses and problems, list of ongoing medication and reasons for discontinuing previous medications, previous and planned procedures and operations, and results of the latest laboratory tests. The DS then examines the data, executes relevant scripts, and sends the output back to the EPR to be shown to the user. The user is able to choose, which type of the output is shown to him/her, and he/she is also able to deactivate specific outputs for individual patients. This will help to prevent showing irrelevant and unnecessary alerts or reminders. Every reminder will have a unique ID that is used for identifying outputs to be blocked. If several scripts produce the same output, it will only be shown once.

The outputs are coded for importance on four levels. The user can choose the lowest importance level for the output he/she wants to see.

When a new trigger input, as described above, is entered to the EPR, the new data including relevant updated basic data is sent to the DS service. Again, relevant scripts are executed and sent back to the EPR.

The DS service can be utilised in virtual health examinations. In this case, the EPR sends data of a specified patient group to the DS service. The DS service performs relevant scripts in a batch run, and sends the feedback to the user as a list of patients and outputs for them. The virtual health examinations help to find patients with specified problems and need for interventions,

but they can also be utilised for quality assessment for specific patient groups and health care processes. A patient-specific structured treatment plan (included in the core data elements listed in Table 1) serves as a storage for outputs produced by the virtual health examination.The physician or nurse must actively accept the suggestions produced by the DS service before they become part of the actual treatment plan.

The third way to utilise the decision support is to automate routine procedures. The DS service can suggest drug and dosing according to the diagnosis, other relevant patient data and treatment guidelines, and send data to the EPR for prescription. The DS service can also collect data for medical certificates or referrals, and send it back to the EPR. Interactive consultation and referral forms will be used. Messages to patients of test results and of preparations for tests and procedures can be drafted by the DS service.

For automated retrieval of relevant guideline documents and patient information, all guidelines have been indexed with diagnostic codes (both ICD-10 and ICPC-2).

Various functional possibilities of the decision support are listed in table 2.

Table 2. Different ways to use the decision support.

Use situation	Output	Examples
On-line feedback	Reminders	Laboratory tests needed
	Alerts	Drug contraindication
	Prompts	Recommended medication
Virtual health examination	Lists	Patients who should be re-evaluated or need a specific intervention
Automatic functions	Guidelines	Guidelines relevant to new diagnosis
	Prescriptions	Drug recommended for diagnosis
	Referrals	Data needed for surgical evaluation

5 Management of the Evidence-Based Data

The evidence and knowledge underlying reminders, prompts and alerts are written in script descriptions (SD), which have a standard format. The format includes the name and ID of the SD, the description of it's function (Appendix 1) and logic, links to evidence summaries that grade the quality of evidence, the reminders, prompts or alerts they produce, discussion of possible harms, references, and meta-data containing keywords, author, publisher and dates of creation and next scheduled update.

The SD's are reviewed by a group of medical specialists. Subsequently, their function and logic are tested before implementation. The feasibility of the decision support will be analysed in qualitative and observational studies, and the effectiveness will be studied in a randomised controlled trial. Seven focus group interviews have been performed and are currently under analysis, and the observational studies are being planned, including e.g. surveys to be repeated as the project proceeds. These will inform planning the RCT. Qualitative assessments will continue to accumulate vital information (why the DS system is / is not working).

Accordingly, decision support for drug therapy will be managed by integrating various structured, nationally maintained medication databases with the EPRs. These include databases for indications, contraindications, interactions, allergy groups and use during pregnancy and breast-feeding. The structured contents of the databases contain codes defined in Table 1 for integrating with EPR data.

Interactive forms are produced using standard Xforms format [6] and XSLT transformation to produce user interfaces for the forms. Integrated care pathways will be composed of national and locally tailored parts.

To ensure the quality of the decision support service, all the databases are centrally managed by the Finnish Medical Society Duodecim.

6 Discussion

In developing a decision support service, the adoption of existing international standards and applications must be considered in the context of national and local clinical, organizational and technical environments.

There is a clear need for international cooperation and standard development in collecting the evidence and developing the clinical contents of the decision support functions. Our solution is to produce script descriptions on the basis of best evidence derived from systematic reviews (such as the Cochrane reviews) and other reliable evidence, and build on the expertise of clinical guideline developers. The script descriptions and evidence summaries are written in natural language, and they are available in a continuously updated, searchable database. We call for international cooperation in developing decision support contents.

On the technical side, the most international efforts should be directed at creating international standards for structuring and coding the patient data in the electronic medical record. If all essential patient data were available in a standard format or convertible into it, and the coding systems and units of measurements could be automatically converted into defined concepts, specified code systems and namespaces, decision support systems could be used in different environments and shared internationally. We have adopted HL7 CDA R2 format for input data, and use an UMLS-based metathesaurus for defining the concepts.

For integration with electronic patient record systems, a web service using international standard interfaces and messages is a practical and flexible solution. After the interfaces have been defined and implemented, the service can be supplied and maintained nationally or internationally as a separate module. The emerging HL7 Decision Support Service specification will probably provide a basis for integration in the future, as exemplified by SEBASTIAN developed at Duke University Medical Center (7).

Considerable international effort has been put into developing a syntax for writing decision support functions. However, widely accepted standards do not seem likely in the near future. Therefore, we have not adopted an existing "decision support language", but developed our own simple object model on the basis of the needs of clinicians authoring the contents of the DS database, and created web-based tools for immediate testing of the DS functions. The executable scripts are written in common and widely used programming language (JavaScript, possibly also in Java or Visual Basic Script if required by applications). Our aim is to produce decision support scripts that can be easily understood and modified by any programmer, and readily integrated in web-based user interfaces.

A website containing a description of the DS service, the object model, script syntax and a sample of script descriptions is available at www.kaypahoito.fi/decisionsupport/decisionsupport.htm.

REFERENCES:

1. http://www.kaypahoito.fi
2. Varonen H, Jousimaa J, Helin-Salmivaara A, Kunnamo I. Electronic primary care guidelines with links to Cochrane reviews--EBM Guidelines. Fam Pract 2005;22:465-9.
3. http://www.terveysportti.fi
4. Garg AX, Adhikari NK, McDonald H, Rosas-Arellano MP, Devereaux PJ, Beyene J, Sam J, Haynes RB. Effects of computerized clinical decision support systems on practitioner performance and patient outcomes: a systematic review. JAMA 2005;293:1223-38.
5. Kawamoto K, Houlihan CA, Balas EA, Lobach DF. Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success. BMJ 2005;330:765.
6. http://www.w3schools.com/xforms/
7. Kawamoto K, Lobach DF. Design, implementation, use, and preliminary evaluation of SEBASTIAN, a standards-based web service for clinical decision support. Proceedings / AMIA Annual Symposium. 2005:380-384.

Appendix 1. Script description in natural language

Selection of diuretics for people with renal insufficiency.

The script checks checks if the patient's drug list includes thiazide diuretics: hydrochlorothiazide alone (ATC C03AA) or in combination (ATC C03EA). If a thiazide compound is present and if serum creatinine (FIN-KL2143) is above 150 µmol/l, reminder (1) is shown.

Reminder (1): The serum creatinine level is increased (>Observed value> µmol/l). Furosemide is recommended instead of a thiazide diuretic.

Appendix 2. Script written in raw script syntax

(The example includes checking the time stamp of serum creatinine, which is not included in the other examples)

```
DSS name="Selection of diuretics for people with renal insufficiency"
id="scr00005"

 Variables
    Thiazides =  Record.Client.Drugs(ATC:C03AA Or
    ATC:C03EA))
    S-Crea =  Record.Client.Observations.Laboratory(KL:2143)
    S-Crea_age =   Now –
    Record.Client.Observations.Laboratory.TimeStamp(KL:2143)
 End Variables

 Messages
     M1(en) = "The patient is on thiazides and there is no recent value for
    serum creatinine. Checking serum creatinine may be considered."

     M2(en) = "The serum creatinine level is increased ([S-Crea)]
    µmol/l).  Furosemide is recommended instead of a thiazide diuretic at
    least when serum creatinine exceeds 200 µmol/l."

     Reminder_1 =      Script.Message.Reminder.Create(M1)
     Prompt_1 =        Script.Message.Prompt.Create(M2)
 End Messages

 Flow
     If Thiazides = True
          If S-Crea = NULL Or S-Crea_age > 1 y
               Reminder_1    // Check S-Crea
               Exit
          End If

          If S-Crea > 150
               Prompt_2      // Switch to furosemide
          End If
     End If
  End Flow
End DSS
```

Appendix 3. Script written in Javascript

```
<?xml version="1.0" encoding="iso-8859-1"?>
<script id="1">
  <!-- Description in real language -->
  <description>
      The script checks if patient's drug list includes thiazide diuretics:
      Hydrochlorothiazide alone (ATC C03AA) or in combination (ATC
      C03EA). If a thiazide compound is present and if serum creatinine
      (FIN-KL2143) is above 150 µmol/l, reminder is shown.
  </description>
<!-- Script source code (not parsed)-->
 <source-code><![CDATA[
 // This function is called by the DS service (a Java application).
    function func() {
   // Variables to be derived from patient data.
        var drug_code_1 = "ATC-C03AA";      // Hydrochlorothiazide
        var drug_code_2 = "ATC-C03EA";      // Hydrchlorothiazide  in
                                            // combination
        var lab_test_code_1 = "FIN-KL2143"; // Serum creatinine
        var thiazidePresent = false         // Is a thiazide diuretic on the
                                            // patient's medication list?
        var serumCreatinineHigh = false;    // Is serum creatinine > 150
                                            // µmol/l
        var serumCreatinineLevel = 0;       // Serum creatinine value
        var serumCreatinineUnit = "";       // Serum creatinine unit

////////////////////////////////////////////////////////////////////////////////////////////////////
   // Checking medications.
   // medications is a variable that is provided for the script as input.
   // The variable is a list containing objects of the Medication class
        if (medications != null) {
            var i = medications.iterator();
            while (i.hasNext()) {
             var medication = i.next();
```

```
// Check if the input ATC code value corresponds to the
// searched value (hydrochlorothiazide, ATC-C03AA).
        if (medication.getCode().equals(drug_code_1)) {
            thiazidePresent = true;
              }
              // The next input ATC code value is checked.
              if (medication.getCode().equals(drug_code_2)) {
                 thiazidePresent = true;
              }
           }
         }
////////////////////////////////////////////////////////////////////////////////////////////////////
// Check laboratory test code values
// The laboratory test results are provided for the script in the same way
// as the medications
        if (labtests != null) {

           // Check only if a thiazide diuretic is on the medication list.
           if (thiazidePresent) {
              var i = labtests.iterator();
              while (i.hasNext()) {
              var labtest = i.next();

             // Check if the input lab test code value corresponds to the
             // searched value serum creatinine, code FIN-KL2143.
                if (labtest.getCode().equals(lab_test_code_1)) {
                // The lab test result was found. Is the value above 150
                // and the unit µmol/l?
                    if (labtest.getResult() > 150 &&
                    labtest.getUnit().equals("µmol/l")) {
                       serumCreatinineHigh = true;
                       serumCreatinineLevel = labtest.getResult();
                       serumCreatinineUnit = labtest.getUnit();
                    }
                 }
              }
           }
         }

////////////////////////////////////////////////////////////////////////////////////////////////////
      // If both codes searched (medication and lab tests) were found
      // and the code values corresponded to the values given in the script,
      // a message is returned
      if (thiazidePresent && serumCreatinineHigh) {
         var reminder = "The serum creatinine level is increased (";
         reminder += serumCreatinineLevel + " ";
         reminder += serumCreatinineUnit + "). ";
         reminder += "Furosemide is  recommended instead of
         thiazide diuretic.";
         return reminder;
      }
   }
 ]]></source-code>
</script>
```