MEDINFO 98
B. Cesnik et al. (Eds)
Amsterdam: IOS Press

Self-Documenting Structured Reports using Open Information Standards

Charles E. Kahn, Jr.

Medical College of Wisconsin, Milwaukee, USA

Abstract

Structured reporting systems use standardized data elements and predetermined data-entry formats to record observations. This article describes a system for structured data entry and reporting that generates reports encoded in the Standard Generalized Markup Language (SGML), an open, internationally accepted standard for document interchange. The structured report is self-documenting: it includes a definition of its allowable data fields and values encoded as a report-specific SGML document type definition (DTD). By linking its reporting concepts with those of external vocabularies such as the UMLS Metathesaurus, this system can create open, universally comprehensible structured reports.

Keywords

Structured Reporting; Standards; SGML; Computer-based Patient Record

Introduction

Structured Reporting

Clinical observations in a computer-based patient record (CPR) may be entered as natural language or as structured data. Structured data entry, the process of recording observations by selecting concepts directly from a standard concept set, offers several potential advantages. Structured reporting systems can capture more complete information than comparable natural-language records [1,2], can facilitate computer-based decision support [3,4], and may reduce the time and expense of dictating and transcribing reports. Standardized observational data acquired directly from the clinician may lower the cost and increase the certainty of data summaries needed for outcomes research.

Another important goal for the emerging CPR is portability [5]. Ideally, the information within a patient's record should be comprehensible to authorized users regardless of the specifics of their local computer system. Open, platform-independent systems should adhere to national and international standards for exchange of medical information.

Standard Generalized Markup Language (SGML)

SGML is an internationally recognized standard for describing the format and content of documents [6-8]. An SGML document consists of a declaration, a Document Type Definition (DTD), and the document instance. The declaration defines the document's lexical properties (e.g., which symbols will serve as special characters). The DTD defines the document's syntactic structure: it specifies the names of allowable elements, how often an element may appear, and the order in which elements must appear. The DTD describes the contents of elements, that is, the names of other elements that are allowed to appear inside them.

Document elements are delimited by "tags," which are keywords enclosed in angle brackets. Elements typically have an opening tag (e.g., `<TITLE>`) and a closing tag (`</TITLE>`). Attributes that further describe the element can appear within the opening tag's angle brackets.

Toward Open, Self-Documenting Structured Reports

Our group has developed a platform-independent, World Wide Web-based interface for structured data entry [9], an SGML-based language for designing structured reporting applications [10], and an interactive Web-based platform for creating structured reporting applications [11]. This article describes a new extension of this system: instead of outline-format text, the system generates an SGML document as its output. The structured report includes a definition of its allowable data fields and values encoded as a report-specific SGML document type definition (DTD). The resulting report is thus both portable and self-documenting.

SPIDER

SPIDER (Structured Platform-Independent Data Entry and Reporting) is a computer-based system for structured reporting. SPIDER is implemented on a UNIX-based system and uses the public-domain NCSA httpd 1.4 server (National Center for Supercomputing Applications, University of Illinois, Urbana, IL). SPIDER's software is written in the Perl programming language (version 5.0).

Data Entry and Reporting Markup Language (DRML)

SPIDER's knowledge base is specified in the Data-entry and Reporting Markup Language (DRML), an SGML-based markup language that was developed to allow physicians to

define the data-entry elements and format of a wide variety of reporting applications [10]. DRML uses hierarchically organized reporting concepts, and provides a means for users to exchange knowledge bases for reporting systems across different hardware and software platforms. Figure 1 shows a simple DRML document.

Figure 1 - DRML document defines a reporting application for renal ultrasonography.

Each DRML document specifies the content and format of the data-entry forms and reports for an entire class of reports. After

```
<title>Renal Ultrasonography
<group header>Clinical Information
    <bin>Pain
    <bin>Hematuria
</group>
<repeat by=row hidden header>Kidneys
    <num units="cm">Size
    <group header option>Hydronephrosis
        <bin>None
          <bin>Mild
        <bin>Severe
    </group>
  <for>
    <rgroup header>Left Kidney
    <rgroup header>Right Kidney
</repeat>
<group header option>Diagnosis
    <bin>Normal
    <bin>Abnormal
</group>
```

the required title, the document contains one or more reporting concepts or formatting codes. Each concept may be a single data element (binary, numeric, or textual), a collection of data elements (GROUP), or a block of repeated data elements (REPEAT). Binary (BIN) items take on a Boolean value of 1 (true) or 0 (false). Numeric (NUM) items allow entry of uni- or multidimensional numeric values. Textual (TXT) items allow entry of one or more lines of narrative-text data.

Reporting concepts can be organized hierarchically in groups. The parent concept is identified by the <GROUP> opening tag. Subconcepts are bounded by the group's opening and closing tags. Each concept in the hierarchy is a modifier of its parent concept and can only be used in conjunction with its parent. Groups can be nested to arbitrary depth. The </GROUP> closing tag must be used to specify the end of the group.

Groups of concepts can be repeated within a "repeat block." The repeat block consists of a <REPEAT> opening tag, a set of items or groups to be repeated, the <FOR> tag, and a list of "repeat group" (RGROUP) elements under which to repeat the concepts; the block is terminated by the </REPEAT> tag. REPEAT blocks cannot be nested.

The opening tags for DRML elements can include attributes that specify how a concept is displayed or used. Concepts can be linked to external vocabularies using the CID attribute, which points to a concept in an external vocabulary or knowledge source such as the Unified Medical Language System (UMLS) Metathesaurus [12]. The CREL ("concept relationship") attribute indicates whether the SPIDER concept is narrower than, broader than, or equivalent to the external concept.

SPIDER User Interface

SPIDER uses the platform-independent technology of the World Wide Web for data entry and report display [9]. SPIDER transforms the concepts specified in the DRML document into familiar graphical objects such as text windows, checkboxes, and radio buttons that are incorporated into Web data-entry forms (Figure 2). The hierarchy of concepts is preserved so that the Web client program can display the form elements with appropriate levels of indentation. Repeat blocks are displayed using HTML's table-formatting capabilities.

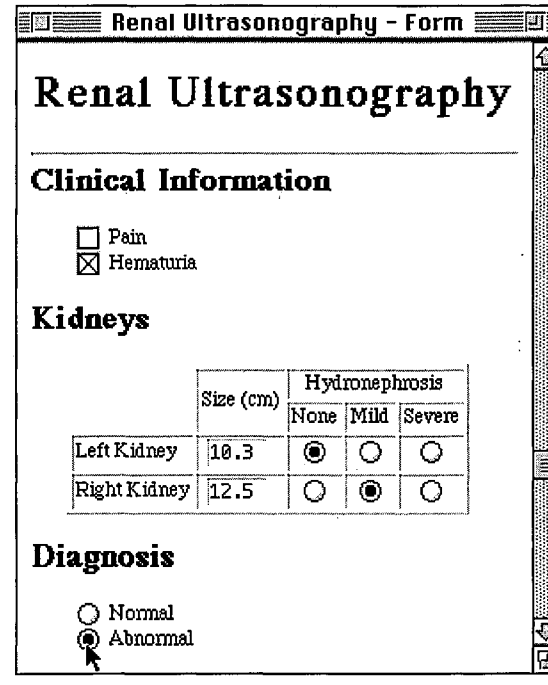

Figure 2 - Data-entry form generated by SPIDER based on the DRML document in Figure 1.

Creating the Report

Once the user has entered information onto the Web-based form and transmitted the form back to the server, SPIDER creates an SGML document containing the information. As with all SGML documents, it consists of an SGML declaration, a DTD, and a document instance. By creating a DTD that expresses the hierarchy, allowable elements, and relationships of the reporting application, SPIDER creates an open, standardized structured report that documents itself.

```
<!--  ============================================================
SPIDER:  Structured Platform-Independent Data Entry and Reporting
generated DTD --> Renal Ultrasonography  (1997.07.03)
Copyright (c) 1997, Charles E. Kahn, Jr., MD
ALL RIGHTS RESERVED

Element     Min     Content                          (Exceptions)
------------------------------------------------------------------ -->

<!ELEMENT   report      O O     (header, x0001)                          >

<!ELEMENT   header      O O     (who what when where)                    >

<!ELEMENT   who         O O     (id name)                                >
<!ELEMENT   id          - O     (#PCDATA)                                >
<!ELEMENT   name        - O     (#PCDATA)                                >

<!ELEMENT   what        - O     (title)                                  >
<!ELEMENT   title       - O     CDATA                                    >

<!ELEMENT   when        O O     (date time)                              >
<!ELEMENT   date        - O     #PCDATA                                  >
<!ELEMENT   time        - O     #PCDATA                                  >
<!ELEMENT   where       - O     CDATA                                    >

<!ELEMENT   x0001       - O     (x0002?, x0005?, x0020?)                 >
<!ELEMENT   x0002       - O     (x0003?, x0004?)                         >
<!ELEMENT   x0003       - O     EMPTY                                    >
<!ELEMENT   x0004       - O     EMPTY                                    >
<!ELEMENT   x0005       - O     (x0006?, x0013?)                         >
<!ELEMENT   x0006       - O     (x0007?, x0008?)                         >
<!ELEMENT   x0007       - O     CDATA                                    >
<!ELEMENT   x0008       - O     (x0010 | x0011 | x0012)                  >
<!ELEMENT   x0010       - O     EMPTY                                    >
<!ELEMENT   x0011       - O     EMPTY                                    >
<!ELEMENT   x0012       - O     EMPTY                                    >
<!ELEMENT   x0013       - O     (x0007?, x0008?)                         >
<!ELEMENT   x0020       - O     (x0022 | x0023)                          >
<!ELEMENT   x0022       - O     EMPTY                                    >
<!ELEMENT   x0023       - O     EMPTY                                    >

<!--    ------------------------------------------------------------------
            Element     Name    Value       Default
        ------------------------------------------------------------ -->

<!ATTLIST   report      id      ID          #REQUIRED                    >
<!ATTLIST   x0001       name    CDATA #FIXED "Renal Ultrasonography"     >
<!ATTLIST   x0002       name    CDATA #FIXED "Clinical Information"      >
<!ATTLIST   x0003       name    CDATA #FIXED "Pain"                      >
<!ATTLIST   x0004       name    CDATA #FIXED "Hematuria"                 >
<!ATTLIST   x0005       name    CDATA #FIXED "Kidneys"                   >
<!ATTLIST   x0006       name    CDATA #FIXED "Left Kidney"               >
<!ATTLIST   x0007       name    CDATA #FIXED "Size"
                        units   CDATA #FIXED "cm"                        >
<!ATTLIST   x0008       name    CDATA #FIXED "Hydronephrosis"            >
<!ATTLIST   x0010       name    CDATA #FIXED "None"                      >
<!ATTLIST   x0011       name    CDATA #FIXED "Mild"                      >
<!ATTLIST   x0012       name    CDATA #FIXED "Severe"                    >
<!ATTLIST   x0013       name    CDATA #FIXED "Right Kidney"              >
<!ATTLIST   x0020       name    CDATA #FIXED "Diagnosis"                 >
<!ATTLIST   x0022       name    CDATA #FIXED "Normal"                    >
```

Figure 3 - SGML document type definition (DTD) for the Renal Ultrasonography reporting application defined in Figure 1.

Report-specific DTD

SPIDER creates a report-specific DTD; this DTD defines a markup language for the particular reporting application, in this case, renal ultrasonography (Figure 3). The REPORT element consists of a HEADER and the top-level reporting concept, X0001. The HEADER consists of mandatory WHO, WHAT, WHEN, and WHERE elements, which may appear in arbitrary order. The WHO element contains elements that specify the patient's ID and name. The WHAT element contains the mandatory report title. The WHEN element specifies the date and time of the report. The WHERE element indicates the institution at which the report was generated. Note that the tag for the WHO element, for example, is optional; it is implied by the presence of its ID and NAME subelements.

The reporting elements are assigned consecutive integers (X0001, X0002, etc.) in order of their appearance in the DRML document. Each element's "content model" describes the elements that appear as its components. The top-level element, X0001 ("Renal Ultrasonography"), has three components: X0002 ("Clinical Information"), X0005 ("Kidneys"), and X0020 ("Diagnosis"). Element X0002 contains elements X0003 ("Pain") and X0004 ("Hematuria"). The notation "`(x0003?, x0004?)`" indicates that the X0002 element's content is X0003 followed by X0004. A question mark indicates that the element may be absent or may appear once. The comma indicates that the sub-elements are to appear in the stated sequence.

For element X0020 ("Diagnosis"), the OPTION attribute in the DRML report-definition file (Figure 1) indicates that exactly one of the element's sub-concepts — "Normal" (X0022 and "Abnormal" (X0023) — may be selected. The data-entry objects for these concepts are displayed as radio buttons (Figure 2). The notation "`(x0015 | x0016)`" for element X0020's content model indicates that either X0022 or X0023 must appear as a sub-element, but not both.

Document Instance

The document instance (Figure 4) encodes the values entered by the user into the markup language defined in the report-specific DTD. The report contains the patient's ID and name, the date, time, place, and title of the report.

The tags for report elements appear only if that element is active in the report. For binary items, that item must be checked. The tags for groups and repeat blocks appear if any of their sub-items are present. For numeric or textual items, the value entered by the user appears following the tag. Thus, for the size of the left kidney, indicated by element X0007 ("Size") within element X0006 ("Left Kidney"), the value "10.3" follows the tag.

Items within a repeat block are defined once in the DTD as elements and appear as sub-elements of the appropriate higher-level elements. Thus element X0007 ("Size") appears as a component of both X0006 ("Left Kidney") and X0013 ("Right Kidney"). Several numbers appear to be missing from the sequence; these numbers are assigned when the data-entry form is generated and are not used for the concepts that appear in the structured report.

```
<report id=867970619>
  <id>123456789
  <name>TEST, PATIENT
  <date>1997-07-03
  <time>10:48:16
  <where>MCW Clinics, Milwau-
kee, WI
  <title>Renal Ultrasonography
  <x0001>
   <x0002>
    <x0004>
  <x0005>
   <x0006>
    <x0007>10.3
    <x0008>
     <x0010>
   <x0013>
    <x0007>12.5
    <x0008>
     <x0011>
  <x0020>
   <x0023>
</report>
```

Figure 4 - SGML document instance generated from the values entered on the data-entry form shown in Figure 2.

Discussion

The suite of tools described here offers a means to creating and implementing structured reporting applications using the open technologies of SGML and the World Wide Web. DRML provides a simple, generalized language for describing a wide variety of reporting applications, including procedural reports (e.g., history and physical examination, diagnostic imaging findings) and structured questionnaires. DRML helps designers concentrate on the content of reports, rather than on user-interface programming, and provides a means for users to exchange knowledge bases for reporting systems across different hardware and software platforms. The use of the World Wide Web for data entry offers a familiar, uniform interface between physicians and the computer system.

SGML-based approaches provide conformance to an open, international standard for document interchange. Investigators have explored the utility of SGML for organizing the computer-based patient record from loosely structured textual documents [13,14]. A prototype voice-enabled structured reporting system that incorporates SGML technology has been developed [15]. The expanding array of public-domain and commercial SGML tools assures that SGML documents will be interpretable across a variety of platforms. In addition to the syntactic comprehensibility afforded by SGML, linkage of SPIDER reporting concepts to UMLS Metathesaurus entries will allow cross-platform conceptual understanding. Further work in this area is underway.

Structured reporting may help satisfy the increasing demands for information by which to judge the quality and effectiveness of medical interventions. The ability to store and retrieve information consistently may lead physicians to more readily embrace structured reporting techniques. Such efforts should help advance the development and use of systems for computer-based patient records.

Acknowledgments

Supported in part by USPHS grant LM05705 from the National Library of Medicine. The author thanks Ondrej Zoltan for computer systems support.

References

[1] Kuhn K, Gaus W, Wechsler JG, et al. Structured reporting of medical findings: evaluation of a system in gastroenterology. *Methods Inf Med* 1992;31:268-274.

[2] Bell DS, Greenes RA. Evaluation of UltraSTAR: performance of a collaborative structured data entry system. In: Ozbolt JG, ed. *Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care.* Philadelphia: Hanley-Belfus, 1994: 216-221.

[3] Campbell KE, Wieckert K, Fagan LM, Musen MA. A computer-based tool for generation of progress notes. In: Safran C, ed. *Proceedings of the 17th Annual Symposium on Computer Applications in Medical Care.* New York: McGraw-Hill, Inc., 1993: 284-288.

[4] Kuhn K, Zemmler T, Reichert M, Rosner D, Baumiller O, Knapp H. An integrated knowledge-based system to guide the physician during structured reporting. *Methods Inf Med* 1994;33:417-422.

[5] Dick RS, Steen EB, ed. *The Computer-Based Patient Record: An Essential Technology for Health Care.* Washington, DC: National Academy Press, 1991.

[6] International Standards Organization. *ISO 8879:1986 Information processing — Text and office systems — Standard Generalized Markup Language (SGML).* Geneva: International Standards Organization, 1986.

[7] Goldfarb CF. *The SGML Handbook.* Oxford: Clarendon Press, 1990.

[8] van Herwijnen E. *Practical SGML.* (2nd ed.) Boston: Kluwer Academic Publishers, 1994.

[9] Kahn CE Jr., Wang K, Bell DS. Structured entry of radiology reports using World Wide Web technology. *RadioGraphics* 1996; 16:683-691.

[10] Kahn CE Jr. A generalized language for platform-independent structured reporting. *Methods Inf Med* 1997; 36:163-171.

[11] Kahn CE Jr., Pfeifer KJ. Interactive construction of structured reporting applications. *RSNA EJ* 1997; 1:article 0028. Available: http://ej.rsna.org/.

[12] Lindberg DAB, Humphreys BL, McCray AT. The Unified Medical Language System. *Methods Inf Med* 1993; 32:281-291.

[13] Essin DJ. Intelligent processing of loosely structured documents as a strategy for organizing electronic health care records. *Methods Inf Med* 1993; 32:265-268.

[14] Essin DJ, Lincoln TL. An information model for medical events. In: Ozbolt JP, ed. *Proceedings of the 18th Annual Symposium on Computer Applications in Medical Care.* Philadelphia: Hanley & Belfus, 1994: 509-513.

[15] Rosenthal DF, Bos JM, Sokolowski RA, et al. A voice-enabled, structured medical reporting system. *J Am Med Informatics Assoc* 1997; 4:436-441.

Address for correspondence

Charles E. Kahn, Jr., MD
Section of Information and Decision Sciences
Department of Radiology
Medical College of Wisconsin
9200 West Wisconsin Avenue
Milwaukee, Wisconsin 53226 USA
E-mail: kahn@mcw.edu
URL: http://www.mcw.edu/midas/spider/