The DICOM (Digital Imaging and Communications for Medicine) standard is a non-proprietary data interchange protocol, digital image format, and file structure for image and image-related information. It is typically used in radiology, cardiology, and similar imaging-intensive departments. DICOM is used in these contexts to integrate and facilitate communication among image-acquisition, waveform, archiving, and information system components. Nevertheless, for applications to handle information from DICOM objects, DICOM tools are required for decoding and encoding the messages. Systems in departments other than these often do not support DICOM but use other proprietary or standard communication protocols, and the number of such systems outnumbers systems that directly support DICOM.

The elusive goal of an integrated electronic medical record is facilitated by object-oriented representations and Web-based interfaces. These will enable physicians to use off-the-shell technologies such as browsers to access patient information. Likely scenarios would include the retrieval by a radiologist of images stored in a picture archiving and communication system and their display for diagnostic interpretation or post-processing, with demographic and study information originally obtained from a hospital information system and radiology information system. At the workstation, the radiologist can then create structured reports that can be mapped from DICOM to open technologies such as XML (Extensible Markup Language). These XML-based...
reports can offer enterprise access to the key study information and related images via a mobile device or a light-weight viewing terminal using a browser or thin client. In this way vital information can be passed seamlessly from system to system, within and across departments, and made available as needed at the point of care, with the aggregated value of hierarchically structured information as opposed to natural language format.

It has been documented that clinicians prefer an outline report with hierarchic standardized vocabularies and structures over a natural language format. Nevertheless, the current actual usage of standard formats for this purpose is minimal at best, with most of the effort going instead into voice recognition and capture of narrative reports. The DICOM Structured Reporting (SR) specification, a supplement to the DICOM standard, is intended to address the structuring of captured data, supporting and structuring conventional free-text reports commonly used in diagnosis. It provides the capability to structure information to enhance the precision, clarity, and value of clinical documents. The DICOM SR specification supports a semantically rich representation of image and waveform content that enables experts to share textual and coded data linked to images and waveforms, as well as knowledge about non-linguistic evidence. The purpose of DICOM SR is to improve the expressiveness, precision, and comparability of documentation of diagnostic images and waveforms, so that critical features can be denoted unambiguously by the observer and retrieved selectively by reviewers. This way, findings can be expressed as textual or coded information, numeric measurement values, and references to spatial or temporal regions of interest.

One main challenge for DICOM SR is to truly interoperate within a health care enterprise, in different clinical scenarios, with different information-exchange standards. Other health care standardization bodies, such as Health Level Seven (HL7), are working on using XML to provide well-structured hierarchies to patient records and also to facilitate the integration of image and non-image medical information into the broader health care context.

We have produced an object-oriented model based on the Unified Modeling Language (UML) that represents the DICOM SR standard Information Object Definition (IOD) hierarchy, macro representation, its characteristics of recursion, and some of the constraints specified in the standard that can be represented by the current state of UML modeling technologies. We have also derived an open exchangeable representation of this model using XML Document Type Definition (DTD) and have identified some of the issues derived from semantic limitations in current XML technologies. We expect developers, analysts, and system architects who are interested in creating applications that are compliant with the DICOM SR specification to benefit from this work.

This document is organized as follows: Section 1 offers a brief introduction to this document. Section 2

---

\*Health Level 7 (HL7) is currently developing the Clinical Document Architecture (HL7 CDA), formerly known as Patient Record Architecture (HL7 PRA), an XML-based structure reference architecture for markup.  
† Integration work is currently under way within the joint effort known as Image Integration Group in HL7 and Workgroup 20 in DICOM.
explains the rationale for modeling the DICOM SR in UML. Section 3 describes the DICOM SR UML modeling decisions. Section 4 describes the XML DTD representation of the UML model. Section 5 presents conclusions, lessons learned, and future work on the subject.

Rationale for Modeling DICOM SR

Some attempts have been made to model the process of creating structured reports by using explicitly stated criteria for making the modeling decisions. Some of these attempts have resulted in concept models that support structured data entry and image retrieval, providing a model for analyzing sets of natural-language reports, although such efforts have typically not been based on industry-supported standards such as DICOM.

For developers who are not DICOM-literate, it is relatively difficult to understand the DICOM SR IODs. Information object definitions in DICOM are based on entity-relationship concepts, although some may argue that they are object-based (i.e., supporting the software engineering concept of encapsulation but not the concepts of inheritance and polymorphism). Interfacing between such relational technologies and object-oriented applications can present a significant semantic and language barrier for application developers and system architects. Furthermore, there is momentum in the DICOM working groups and the health care vendor community to standardize the XML rendition of DICOM SR to allow the extension of structured reporting capabilities to the health care enterprise as a whole.

Unified Modeling Language is a way of specifying, visualizing, constructing, and documenting the artifacts of software systems as well as business models and other non-software systems. We have followed the conventional UML notation and syntax, basing our models on class diagram structures. This notation uses the basic principles of object-orientation to model system structure and behavior. It defines classes and class responsibilities with object-oriented analysis and design concepts such as objects, classes, stereotypes, and relationships.

Mapping the SR relational information model to an object-oriented information model, with the assistance of standard off-the-shelf tools, is an indispensable step toward a standard XML DTD for the DICOM SR specification. Also, XML will ease access to imaging, demographics, and waveform information using Web-based open component technology, addressing interoperability and system integration issues. We expect this step to carry the domain-specific DICOM format into a more friendly and interoperable data format such as XML, which will offer a wider use of relevant data in a variety of multimedia and application settings in which images and reports are viewed in hospitals. Such a model will also provide a useful framework for the interoperability and mapping efforts between HL7 and DICOM being carried on by DICOM Workgroup 20 and HL7 image integration groups.

Modeling Decisions

DICOM SR is intended to support the interchange of expressive compound reports in which the critical features shown by images and waveforms can be unambiguously annotated by the observer, indexed, and retrieved selectively by subsequent reviewers. As stated before, DICOM has been designed to rely on explicit and detailed entity-relationship models. DICOM IODs define the data structures that describe information objects, or logical representations of real-world objects, such as patients and images, involved in radiology operations. The entity-relationship diagram for the radiology department function serves as the basis for DICOM models, showing both the data items required in a given scenario and the interactions and relationships between such items, as shown in Figure 1.

DICOM SR introduces DICOM Services and IODs used for the transmission and storage of structured reports. DICOM IODs are representations of real-world entities (e.g., images and reports) represented in the specification as templates of attributes. DICOM Services can be composite and normalized. Composite services are focused on storage, query, retrieval, and transfer of data and are optimized for image and interpretation data interchange. Normalized services are designed to support a wider range of information management functionality and are focused on basic information management functionality (create, delete, update, and retrieve and a notification service).

The DICOM SR Service-Object Pair (SOP) definitions allow users to link text and other data to particular images and waveforms and to store the coordinates of findings so that they can see exactly what is being described in a report. These DICOM SR IODs and corresponding DICOM SR Storage SOP Classes enable the query and retrieval of SR SOP Instances as Instance-level entities, following the DICOM Query/Retrieve model. DICOM SR IODs are
grouped in Information Entities (IEs), which contain IOD modules (Table 1).

Information object definition modules contain attributes, which in turn may refer to other attributes or to attribute groupings, called Macros. The following descriptions define the modeling decisions and mapping rules for the different DICOM SR elements made toward UML and XML DTD representations of the specification, starting from the SRDocument IOD as the root.

**Information Entities and Information Object Definition Module Mapping Decisions**

There are five IEs in DICOM SR: Patient, Study, Series, Equipment, and Document. We have mapped each IE into a class with the same name as the IE to which it refers. An exception to this is the Patient IE, which is mapped into the Patients class, since a Patient IOD module already exists in the next sublevel (Table 2).  

<table>
<thead>
<tr>
<th>DICOM SR Concept</th>
<th>UML Model Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Information entity</td>
<td>Class</td>
</tr>
<tr>
<td>Information object definition module</td>
<td>Class</td>
</tr>
<tr>
<td>Macro</td>
<td>Class</td>
</tr>
<tr>
<td>Attribute</td>
<td>Attribute</td>
</tr>
<tr>
<td>Constraint: Type 1, 2 (mandatory)</td>
<td>Required</td>
</tr>
<tr>
<td>Type 1C, 2C (conditional)</td>
<td>Optional/required</td>
</tr>
<tr>
<td>Type 3 (optional)</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Each IE contains one or more Modules. These modules are mapped into classes with the same name as the IOD to which it refers, except that spaces are removed from the composite IOD names. There are nine IOD Modules within the SR IEs: Patient, Specimen Identification, SR Document Series, General Study, Patient Study, General Equipment, SR Document General, SR Document Content, and SOP Common. For instance, the Patient Study Module is mapped into the PatientStudy class.
Macros and Attributes

Each DICOM Macro is mapped into a class with the same name as the Macro to which it refers, except that spaces and the “Macro” postfixes are removed from its name. For example, the SOP Instance Reference Macro becomes the SOPInstanceReference class.

For the attributes, each attribute is mapped into a class attribute following these rules:
- Change all uppercase letters to lowercase.
- Replace the blank space between two letters with an underscore.
- Remove apostrophes and brackets.
- Replace hyphen (-) and slash (/) with underscore (_).

For example, the SR Document General Module attributes are mapped in the UML model as shown in Figure 2.

Each Sequence attribute is mapped into a class attribute. This class attribute is of a class type that contains the sublevel attributes of the Sequence.

Data Types

DICOM defines a value representation for each attribute. Such values will be used for the atomic attributes. For the composite attributes, such as sequence type, their associated classes serve as their types (Table 3).
Recursion

DICOM SR shows some particular characteristics of recursion, as are present in the SR Document Content module via the Document Relationship Macro, which instantiates itself under certain conditions. Figure 3 shows the relationship of SR Documents to Content Items and the relationships of Content Items to other Content Items and to Observation Context.

The issue of recursion in the Document Content module is a key property that allows multiple containment within structured reports, an important property in numeric measurement-intensive reporting such as ultrasound applications. The DICOM SR specification reflects this complex property by the cross-referencing of DICOM Macros. This representation makes it hard for those who are not DICOM-literate to understand this property. We have approached this problem by modeling the reference relationship to Content Item, as well as its relationship by containment, to reflect this reciprocal recursion. This is a key difference between the various SR SOP Classes as defined in the specification (Figure 4).

Constraints

DICOM SR is rich in constraints. This DICOM SR object model has been created using the Rational Rose 98 Enterprise Edition UML tool set (http://www.rational.com). The DICOM Type 1 and

<table>
<thead>
<tr>
<th>DICOM Data Type</th>
<th>Short Meaning</th>
<th>UML Primitive Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>AE</td>
<td>Application Entity</td>
<td>String</td>
</tr>
<tr>
<td>AS</td>
<td>Age String</td>
<td>String</td>
</tr>
<tr>
<td>AT</td>
<td>Attribute Tag</td>
<td>Unsigned long</td>
</tr>
<tr>
<td>CS</td>
<td>Code String</td>
<td>String</td>
</tr>
<tr>
<td>DA</td>
<td>Date</td>
<td>String</td>
</tr>
<tr>
<td>DS</td>
<td>Decimal String</td>
<td>String</td>
</tr>
<tr>
<td>DT</td>
<td>Date Time</td>
<td>String</td>
</tr>
<tr>
<td>FL</td>
<td>Floating Point Single</td>
<td>Float</td>
</tr>
<tr>
<td>FD</td>
<td>Floating Point Double</td>
<td>Float</td>
</tr>
<tr>
<td>IS</td>
<td>Integer String</td>
<td>String</td>
</tr>
<tr>
<td>LO</td>
<td>Long String</td>
<td>String</td>
</tr>
<tr>
<td>LT</td>
<td>Long Text</td>
<td>String</td>
</tr>
<tr>
<td>OB</td>
<td>Other Byte String</td>
<td>String</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>DICOM Data Type</th>
<th>Short Meaning</th>
<th>UML Primitive Data Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>OW</td>
<td>Other Word String</td>
<td>String</td>
</tr>
<tr>
<td>PN</td>
<td>Person Name</td>
<td>String</td>
</tr>
<tr>
<td>SH</td>
<td>Short String</td>
<td>String</td>
</tr>
<tr>
<td>SL</td>
<td>Signed Long</td>
<td>Signed long</td>
</tr>
<tr>
<td>SS</td>
<td>Signed Short</td>
<td>Signed short</td>
</tr>
<tr>
<td>ST</td>
<td>Short Text</td>
<td>String</td>
</tr>
<tr>
<td>TM</td>
<td>Time</td>
<td>String</td>
</tr>
<tr>
<td>UI</td>
<td>Unique Identifier</td>
<td>String</td>
</tr>
<tr>
<td>UL</td>
<td>Unsigned Long</td>
<td>Unsigned long</td>
</tr>
<tr>
<td>UN</td>
<td>Unknown</td>
<td>String</td>
</tr>
<tr>
<td>US</td>
<td>Unsigned Short</td>
<td>Unsigned short</td>
</tr>
<tr>
<td>UT</td>
<td>Unlimited Text</td>
<td>String</td>
</tr>
</tbody>
</table>

**Figure 3** DICOM SR relationship information model.

**Figure 4** Document relationship macro, showing relationship by containment.
2 attributes are mapped as Required. Type 3 attributes are mapped as Optional. For Type 1C and 2C attributes, which are required under certain conditions, the following rules are used:

- If the conditionality is based on a single constraint, related to the presence of the class to which the attribute belongs, it is mapped as Required.
- If the conditionality is based on multiple constraints, or a single constraint not related to the presence of the class to which the attribute belongs, it is mapped as Optional.

For instance, within the Code Sequence Macro, Code Value is Required only if a Code Sequence, the class to which it belongs, is present, so we represent it as Required. On the other hand, within the same Code Sequence Macro, Coding Scheme Version is Required only if a Code Sequence is present AND the value of Coding Scheme Designator is not sufficient to identify the Code Value unambiguously, so we represent it as Optional. The conditions related to types 1C and 2C cannot be captured by the current version of the UML modeling tool. Additional constraints to the UML model may be represented by new modeling technologies and artifacts (e.g., the Object Constraint Language initiative), by which finer constraints and conditions can be represented in modeling and can be a subject of interest for future versions of this model.

### XML Representation of the UML Model

Early attempts to represent medical information contained in structured reports focused on allowing platform-independent representations of structured reports using open technologies. A new approach, which we consider the most likely path the industry will follow in the near future, is the representation of structured reports using XML, a more efficient and approachable subset of SGML.

We have generated an XML DTD based on this DICOM SR UML model using the following rules and modeling decisions:

- UML classes are mapped to the XML DTD Elements.
- UML class attributes are mapped to XML DTD Elements.
- All UML Association and Uses relationships are mapped to the XML DTD Elements as relationships by containment.
- Each atomic attribute is mapped to an element, which contains five attributes: codingScheme, codeId, type, value and label.

For instance, the DocumentContent class (within the SRDocumentContent class) maps to XML DTD as shown in Figure 5.
On the other hand, a more complex relationship, such as the recursive relationship between Document Content and Document Relationship, within Document Relationship, maps to the XML DTD as shown in Figure 6.

In our experience, once the UML object model has been developed, it is fairly easy to generate an XML DTD representation. This was done by manually mapping the class structure to the DTD framework, taking the model as a reference.

**Conclusions and Future Work**

The promotion of DICOM SR capabilities (at the basic and enhanced SOP Class levels), were an important part of the Integrating the Healthcare Enterprise (http://www.rsna.org) Year 2 demonstrations, which were jointly sponsored by the Radiological Society of North America (http://www.rsna.org) and the Healthcare Information and Management Systems Society (http://www.himss.org) in November 2000.
and February 2001, respectively. We expect our XML representation of DICOM SR, based on this UML diagram, to be close to what the DICOM working groups and the industry will adopt and eventually standardize.

We believe that this object-oriented approach, using XML-based open technologies to interface DICOM binary and HL7 ASCII information for enterprise implementation, will enable relatively simple transfer to various clinical specialties as well as ease the leverage of Web-aware applications and technologies.

During this modeling exercise we encountered a number of obstacles. Probably the most difficult was finding the most direct, object-oriented way to represent the concept of recursion.

One of the initial decisions that we later changed was modeling the DICOM concept of Macros using intermediate logical artifacts, stating them as abstract classes that other implementation classes used. Since the concept of Macro is an artificial DICOM construct that exists as a notation only, we decided at the end not to take this concept into the model.

We have demonstrated that a complete XML DTD can be easily produced from the DICOM specification once a good understanding and representation of the system is achieved. The UML proved to be an extraordinary tool to achieve this level of understanding and representation.

References

Call for Papers

Special Issue on Simulation in Medical Informatics

Computer simulation provides a flexible and generalizable methodology for planning and evaluating applications of information technology in health care. Simulation software is increasingly flexible, powerful, and easy to use and understand. Graphic, statistical, and animated output permit quantitative and qualitative answers to “what if” questions. The new generation of personal computers coupled with software with iconographic programming greatly facilitates exploratory studies. This visual approach to model building frees the investigator from dealing with complex mathematical expressions and unfamiliar programming languages. Once a model is developed and validated, it can be run showing the consequences of modifications.

The success of many complex medical informatics applications depends on not only the information system but also the organizational environment in which they are implemented. When an application cannot be evaluated with traditional experimental methods, computer simulation provides a flexible approach to evaluation. Once validated, the model can be used to study the effects of variation in system inputs, differences in initial conditions, and changes in the structure of the system.

Simulation may also be a useful tool for training health professionals. The simulation of patients and clinical situations can provide professionals with training opportunities rarely experienced in real-world situations.

The editors invite the submission of papers that apply simulation to applications of information technology in health care. Examples of applications include but are not limited to the following:

- Organizational simulation of information processing
- Simulation of telehealth networks
- Simulation in the design of information systems
- Simulation of physician order entry systems
- Simulation of the value of health information systems
- Simulation of the use of information systems to prevent medical errors
- Simulation to train health care workers to use information systems
- Use of simulated patients in medical informatics training
- Use of patient simulations in evaluation

Authors are asked to e-mail the editor of the special issue, notifying him of their intent to submit and including a brief abstract of their paper. Potential authors who are uncertain about the appropriateness of their submission for the special issue are invited to send a detailed outline or draft of their paper for preliminary review. Questions should be addressed to:

James G. Anderson, PhD
Editor of the Special Issue
Tel: 765-494-4703
Fax: 765-496-1476
e-mail: Andersonj@sri.soc.purdue.edu

Manuscripts for this issue should be submitted to the JAMIA editorial office by February 15, 2002.

Instructions for authors are available at http://www.jamia.org/misc/ifora.shtml. They include detailed information about acceptable formats for reporting various types of work and instructions for submitting manuscripts via FTP to the Journal’s editorial server.