The New Navigators: from Professionals to Patients
R. Baud et al. (Eds.)
IOS Press, 2003

DicoSE - A web-interface based on XML for visualizing and searching the topology of the data structure defined by the DICOM 3 standard

Michael Prinz, Georg Fischer, Ernst Schuster

Dept. of Medical Computer Sciences, Section on Med. Computer Vision, University of Vienna, Austria

Abstract

To adequately visualize and interpret the data fields of DICOM 3 datasets the data structure defined in the DICOM 3 standard has to be applied. The DICOM 3 data structure is very extensive and therefore costly to implement. We are working on an open source system which provides the data structure via a Java-programming interface. The data is held in a freely available XML-database. As a spin-off we are providing the web-based Dicom Search Engine (DicoSE) which will be available via internet soon. DicoSE allows for searching the DICOM standard data dictionary for defined data fields and visualizes the topology of the data which is present in DICOM datasets acquired by various types of modalities. Thus, the interpretation of the meaning of data fields is supported. For maintaining the data stored in the database a web-based administration interface is provided.

Keywords:

DICOM; XML; Java; search engine

1. Introduction

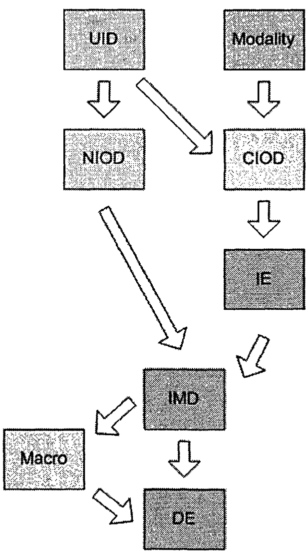
The data fields in DICOM datasets[1] are specified sequentially one after the other without preserving their topology and specifying their meaning. Without a powerful DICOM-viewing application at hand the meaning and the topology of the data are interpretable only by looking up the definitions in the DICOM standard parts 3, 5 and 6 [1]. To support the implementation of DICOM interpreting applications, especially the implementation of our own image archiving and image processing system Melange [2][3][4], we are working on an open source Java programming library for querying the data structure defined by the DICOM standard. By providing the numerical tag values found in the datasets, its meaning, data type, owner and multiplicity of occurrence as stated in the DICOM-standard is returned. For maintaining, visualizing and searching the data residing behind the programming library we implemented the web-interface DicoSE which is the only publicly available DICOM-interface of this kind. There are a lot of DICOM-viewing applications which allow for displaying the image- and textual data in DICOM-datasets but none of them is capable of visualizing and querying the overall structure of the DICOM-standard independently of actual datasets.

[1]The DICOM 3-standard is defined and upgraded annually by a comittee formed by the American College of Radiology (ACR) and the National Electrical Manufacturers Association (NEMA).

2. Methods and Materials

2.1. Topology of the DICOM data

The DICOM data as defined in the DICOM-standard is organized hierarchically (figure 1). The dicom elements (DE) represent the data fields which are listed in datasets acquired by DICOM modalities, e.g. “patient’s address”. They are defined in the DICOM data dictionary (DICOM standard part 6). The various DEs are comprised in Information Module Definitions (IMD), e.g. “patient identification” which in turn are comprised in Normalized Information Object Definitions (NIOD), e.g. “patient”, or in Information Entities (IE) etc. The adequate hierarchy tree to apply can be identified by the Unique Identifier (UID) which should be present in the DICOM dataset. If there is no UID present, the modality tag (a DE-element) may be used for finding an adequate Composite Information Object Definition (CIOD).



Legend:

	Element	DICOM Document
UID	Unique Identifier	PS 3.6-2001 (A)
MOD	Modality	
CIOD	Composite Information Object Definition	PS 3.3-2001 (A)
IE	Information Entitie	PS 3.3-2001
NIOD	Normalized Information Object Definition	PS 3.3-2001 (B)
IMD	Information Module Defintion	PS 3.3-2001 (C)
Macro	Macro	PS 3.3-2001
DE	Data Element	PS 3.6-2001 (5)

Figure 1: The topology of DICOM-data.

2.2. The DicoSE-service

The implementation of the DICOM standard implies the administration of a large amount of data. This can be achieved only by administrating the data in a database. One has to bear in mind that the data has to be maintainable since the standard is updated annually by the DICOM standards committee. We decided to base the data structure on XML and to use the XML-database Xindice [7] which is distributed and freely available under the Apache software license. It allows for directly storing and querying XML-documents through XPath-queries via a Java programming library. It has to be mentioned that Xindice is still work in progress. There have been some efforts in contributing XML-based data structures to the DICOM standard already especially for structured reporting [5][6]. Thus, it seems sensible to rely on XML-based records. For accessing the database the open source EJB-server JBoss [8] was installed on a Linux server. EJB-beans were implemented for accessing the system by client applications. For using the system via the DicoSE-web-interface the Tomcat-Web-server included in the JBoss distribution is used and Java-Server-

Pages were implemented to handle the communication between the web-interface and the EJB-server. There are two different web-access points, one for merely querying the DicoSE-system and one for the maintenance by administrators. The accession of the administration interface is possible only by authorizing as administrator via a login dialog.

2.3. Accessing the data

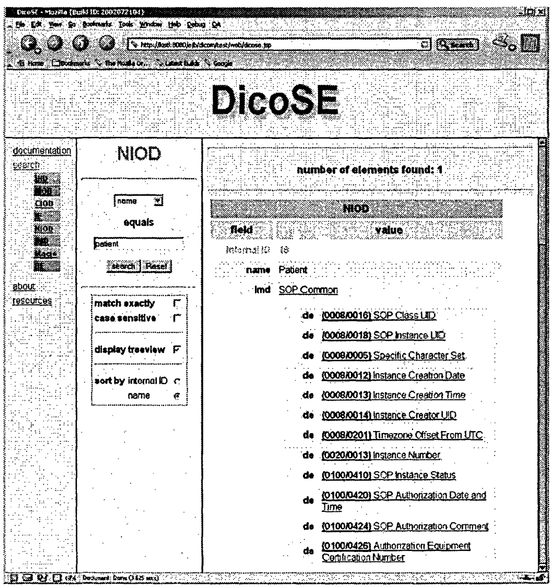


Figure 2 DicoSE web page with the text field for entering the DICOM data to look for. Here the NIOD with the definition name "patient" is queryied and the result is shown on the browser's right side.

The DicoSE-service is accessible with any web browser. After choosing the type of the DICOM information object to be retrieved (figure 1), the DICOM object is searched by entering its name or ID (only available for DEs, UIDs and Modalities) in a text field of an HTML-form. This data is transferred to the Web-Server of the DicoSE-service which calls the corresponding EJB-bean method at the EJB-server via Java Server Pages. The EJB-bean accesses the XML-database via XPath queries to retrieve the data structure defined by the DICOM standard. The located data is returned to the EJB-bean as an XML-document which then is converted to an HTML-document via XSLT-macros and returned to the Web-Server. The Web-Server sends the HTML-document back to the calling browser for presentation (figure 3).

The DICOM information objects are stored as XML-documents (e.g. figure 4) in the Xindice database. Subdocuments (e.g. IMDs in NIOD objects) are referenced by their ID (<ref>-tags in the XML-document). For presentation via the web browser the references are resolved in the querying and retrieving phase and the data of the subdocuments are fully included into the referring document.

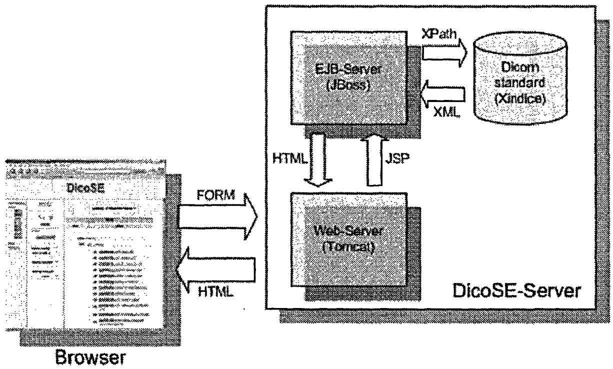


Figure 3: Data transfer chain from the browser to the DICOM-database.

```
<?xml version="1.0"?>
<imd>
    <base>
        <internalId>4</internalId>
        <name>Patient Medical</name>
    </base>
    <subList>
        <de>
            <ref>1142</ref>
         </de>
        <de>
            <ref>3391</ref>
         </de>
        <de>
            <ref>2500</ref>
         </de>
        <de>
            <ref>2421</ref>
         </de>
        <de>
            <ref>2365</ref>
         </de>
        <de>
            <ref>2329</ref>
         </de>
        <de>
            <ref>473</ref>
         </de>
        <de>
            <ref>3</ref>
         </de>
    </subList>
</imd>
```

```
<?xml version="1.0"?>
<de>
    <base>
        <internalId>3391</internalId>
        <id>0010/2110</id>
        <name>Contrast Allergies</name>
        <group>0010</group>
        <element>2110</element>
        <version>3</version>
        <valueRep>LO</valueRep>
        <valueMult>1-n</valueMult>
        <owner />
        <keyword>
            ContrastAllergies
        </keyword>
        <private>false</private>
    </base>
</de>
```

Figure 4: The information objects are stored in the database as XML-records. On the left side the XML-record for the IMD "Patient Medical" is shown. Only references to the included DEs are specified. On the right side one of the referenced XML-records (DE "Contrast Allergies") is shown.

After retrieving the corresponding document, it is converted from XML- to HTML-format and presented by the browser (figure 5). Sub objects (e.g. IMDs in NIODs) are displayed in short form. Thus, the list remains easy to survey. To display the full form of the subdocument, the user only needs to click on the object's name.

NIOD		
field	value	
internal ID	15	
name	Patient	
imd	SOP Common	
	de	(0008/0016) SOP Class UID
	de	(0008/0018) SOP Instance UID
		...
	de	(0100/0426) Authorization Equipment Certification Number
imd	Patient Relationship	
	de	(0008/1110) Referenced Study Sequence
	de	(0008/1150) Referenced SOP Class UID
	de	(0008/1155) Referenced SOP Instance UID
	de	(0008/1125) Referenced Visit Sequence
	de	(0008/1150) Referenced SOP Class UID
	de	(0008/1155) Referenced SOP Instance UID
	de	(0038/0004) Referenced Patient Alias Sequence
	de	(0008/1150) Referenced SOP Class UID
	de	(0008/1155) Referenced SOP Instance UID
imd	Patient Identification	
	de	(0010/0010) Patient's Name
	de	(0010/0020) Patient ID
	de	(0010/0021) Issuer of Patient ID
	de	(0010/1000) Other Patient IDs
	de	(0010/1001) Other Patient Names
	de	(0010/1005) Patient's Birth Name
	de	(0010/1060) Patient's Mother's Birth Name
	de	(0010/1090) Medical Record Locator
imd	Patient Demographic	
	de	(0010/1010) Patient's Age
		...

Figure 5: The (shortened) result of querying the NIOD “patient”. The comprised IMDs and DEs are listed with their names and IDs.

3. Discussion

The implemented search engine DicoSE eases the interpretation of DICOM data when there is no powerful DICOM-viewer at hand. It facilitates obtaining a general view of the topology of DICOM data. Since the data will be used for the image archiving and processing system Melange the administration interface heavily simplifies the maintenance of the DICOM-data in the database.

The functionality of the XML-database Xindice is sufficient for this application. Since the database is not visible externally, Xindice’s lack of accession restriction mechanisms poses no problem. The representation of the data in XML-form heavily simplifies the handling and output of the data since there are a lot of freely available Java programming libraries for handling XML-data at hand (e.g. JDOM, Xerces, Xalan).

For easing the development of applications for handling DICOM-datasets it would be very useful to distribute and maintain the DICOM-standard electronically via well-formed XML documents. The method of providing the standard documents in printed and PDF-version only, makes it very costly and inefficient to implement applications for handling DICOM-datasets.

4. Conclusion

We implemented a web-based and XML-based search engine for the data structure defined by the DICOM 3 standard. It is enhanced by an administration interface for maintaining the used DICOM data. The data is stored as XML-documents in a native XML-database. The cooperation of a browser, a web-server, an EJB-server and an XML-database seems to represent an efficient way (regarding the implementation and the maintenance) to solve the posed requirements.

5. References

[1] National Electrical Manufacturers Association (NEMA). Digital Imaging and Communications in Medicine (DICOM). http: //medical.nema.org. 2001.

[2] Prinz M, Fischer G, Schuster E. Austausch, Archivierung und Verarbeitung medizinischer Bilder mit Melange. *conference of the Österreichische Wissenschaftliche Gesellschaft für Telemedizin, a-telmed 2001*, Innsbruck, Austria. 2001.

[3] Prinz M, Lorang T, Gengler G, Schuster E. Implementation of a Medical Database for DICOM Datasets. *International Health Information Conference, INFOCUS 2000.* Vancouver, Canada. 2000.

[4] Prinz M, Gengler M, Lorang T, Schuster E. MELANGE: A Java-based environment for biomedical image archiving and image processing, *Computer Assisted Radiology and Surgery. CARS 2000. 14th International Congress and Exhibition.* San Francisco, USA. 2000.

[5] Tirado-Ramos A, Hu J, Lee KP. Information Object Definition-based Unified Modeling Language Representation of DICOM Structured Reporting. *J. Am. Med. Inform. Assoc. 2002*; Jan-Feb; 9(1); pp. 63-71.

[6] Van Nguyen A, Avrin DE, Tellis WM, Andriole KP, Arenson RL. What Digital Imaging and Communication in Medicine (DICOM) could look like in common object request broker (CORBA) and extensible markup language (XML). *J. Digit. Imaging. 2001*; Jun; 14 (2 Suppl 1); pp. 89-91.

[7] Apache Xindice, XML-database, http: //xml.apache.org/xindice

[8] JBoss, EJB-application server, http: //www.jboss.org

6. Address for correspondence

Michael Prinz
Dept. of Med. Computer Sciences
Section on Med. Computer Vision
Spitalgasse 23
A-1090 Vienna / Austria
email: Michael.Prinz@akh-wien.ac.at
tel: (+43)(1) 40400-6655
fax: (+43)(1) 40400-6656