Using Grid-Enabled Distributed Metadata Database to Index DICOM-SR

Ignacio BLANQUER¹, Vicente HERNANDEZ¹, José SALAVERT¹, Damià SEGRELLES¹

¹Instituto de Aplicaciones de las Tecnologías de la Información y de las Comunicaciones Avanzadas (ITACA), Universidad Politécnica de Valencia, Camino de Vera S/N, 46022 Valencia Spain (phone: +34 96 387 7007 ext. 88254; Fax: +34 96 387 7274; e-mails: iblanque@dsic.upv.es; vhernand@dsic.upv.es; josator@fiv.upv.es; dquilis@itaca.upv.es

Abstract. Integrating medical data at inter-centre level implies many challenges that are being tackled from many disciplines and technologies. Medical informatics have applied an important effort on describing and standardizing Electronic Health Records, and specially standardisation has achieved an important extent on Medical Imaging. Grid technologies have been extensively used to deal with multi-domain authorisation issues and to provide single access points for accessing DICOM Medical Images, enabling the access and processing to large repositories of data. However, this approach introduces the challenge of efficiently organising data according to their relevance and interest, in which the medical report is a key factor. The present work shows an approach to efficiently code radiology reports to enable the multi-centre federation of data resources. This approach follows the tree-like structure of DICOM-SR reports in a self-organising metadata catalogue based on AMGA. This approach enables federating different but compatible distributed repositories, automatically reconfiguring the database structure, and preserving the autonomy of each centre in defining the template. Tools developed so far and some performance results are provided to prove the effectiveness of the approach.

Keywords. DICOM, Structured Report, DICOM-SR, metadata catalog, AMGA

1. Introduction

The Digital Imaging and Communications in Medicine [1] (DICOM) standard was created by the National Electrical Manufacturers Association (NEMA) to aid in the distribution and viewing of medical images, such as ultrasound, MRI, CT and PET scans, and in the integration of these images in Hospital Systems. Throughout the years, DICOM has developed new features and it is also used to manage other information types such as signals, videos, waveforms and even structured reports through DICOM-SR. DICOM Structured Report [2] (DICOM-SR) proposes a standard way to code and structure radiology reports, including image references and processing results. These reports contain information about medical observations and conclusions that a doctor may find on a diagnosis process related with DICOM studies. DICOM-SR has been adopted into the DICOM standard in 1999 as Supplement 23 [3].

Nowadays, DICOM information systems like Picture Archiving and Communication Systems (PACS) or Radiology Information Systems (RIS) manage
DICOM objects and related patient data efficiently in medical environments for healthcare delivery. All these systems, along with standards as Health Level Seven (HL7), openEHR or EN13606, as well as initiatives such as Integrating the Healthcare Enterprise (IHE) constitute an important effort in organising and structuring medical data within the departments of medical organisations.

Those protocols and systems have enabled medical centres to hold huge amounts of DICOM data. However, except for specific Computer Aided Diagnosis procedures, radiology reports have been poorly coded where standardisation has been possible.

All this DICOM information (specifically DICOM-SRs) that is distributed in different medical centres could be very useful for research and training. Exploiting the information through data-mining techniques as a whole could lead to unrelated knowledge. However, protocols and systems used at intra-hospital level are not sufficient when sharing and managing medical data across different networks and are administrative domains since these data organised for healthcare delivering.

In the last years, many projects aiming at the problem of sharing medical information among different centres have appeared. Several of them target the specific problem of sharing DICOM data. Most projects are based in Peer to Peer Models (P2P) or Grid Technologies. Sharing data among different centres introduces the problem of data security, this issue is a key requirement for medical applications. The system should deal with the different national legal regulations and procedures accepted by the medical community [4]. Today, Grid models could provide acceptable security issues in biomedical applications in Grid environments as [24].

Current projects provide data annotation and data mining on DICOM metadata. DICOM-SR provides elements like spatial and temporal coordinates to indicate diagnostics annotations, containers of data to explain observations, specific codes to indicate a localization, text values to describe observations, number values to measurements, etc. In [2] there is a precise description of DICOM-SR. The information contained in these elements is essential to index medical images in order to perform searches based on the diagnostic information, opening the door to data-mining on the DICOM-SR content from existent DICOM databases in different medical centres.

This paper is centred in the problem of sharing DICOM data, including annotation data from DICOM-SR through standard components that can be integrated in existing Grid infrastructures. This work proposes sharing DICOM data and DICOM-SR content maintaining the tree structure of DICOM-SR. This structure enables federating different but compatible diagnostic structures and speed-up complex queries.

This paper is organized as follows. Second section describes the objectives of the paper. Third section outlines related works on sharing DICOM data, particularly the DICOM-SR content in distributed systems. Fourth section describes the infrastructure needed to accomplish the objectives. Fifth section outlines the data workflows and methodologies employed. Finally, results, conclusions and future works are presented.

2. Objectives

The main goal of this paper is to present a Grid solution that integrates DICOM data and DICOM-SR content among multi-centre data storages, creating knowledge repositories for research and training. DICOM-SR content data are organised according to templates, which define the structure of DICOM-SR reports. This global objective requires solving the following issues:
• To define a specification and a procedure to create DICOM-SR templates. These templates organize the DICOM-SR content data in a tree-like structure and will be the base to organize the information in Grid environments.

• To use Grid standard components to share and store DICOM data. The solution presented in this paper uses the gLite [12] Middleware. Storage Elements (SE) are used as backend to physically store DICOM objects, Local File Catalogs (LFC) reference DICOM objects through logic names, and the ARDA metadata Grid Application (AMGA) organizes DICOM-SR content information and their links to full reports kept in the backends along with the associated DICOM images.

• To develop a DICOM-SR tool that is able to validate an XML description of a DICOM-SR according to a given structure template and generate a DICOM-SR binary file to be uploaded in the Grid infrastructure backend (LFC + SE). This component also stores part of the DICOM-SR content data into an AMGA repository, according to the tree-like relations of the nodes. Database queries will return identifiers of the DICOM-SR objects.

3. Related Works

Mainly, two approaches are used on the literature to share DICOM data among different medical centres: Peer-to-Peer (P2P) and Grid [5] technologies.

In [6] a P2P system that enables a community of radiologist to share DICOM images and their associated diagnosis is presented. This approach shares the diagnosis information in plain text, which could lead to incomplete results.

A system for sharing meta-patient records in a Peer-to-Peer environment called SIGMCC [7] project shows a hybrid P2P architecture based on a Super Peer model [8]. This project uses the concept of Electronic Patient Records (EPRs) to share data among different medical centres by using a mediator-based approach, which defines a global scheme, and the queries against the global scheme are mapped to queries defined on the local data. DICOM introduces new types of data items, so security must be considered.

On the other hand, many scenarios in medical research are suitable for Grid computing. Processing and storing large amounts of data demand large computing power and storage. Data grid technologies such as Storage Resource Broker [9] (SRB), Globus toolkit [10], Unicore [11] or gLite [12] provide general-purpose tools for storing large amounts of data in distributed environments. DICOM connectivity is provided through the development of DICOM plug-ins for standard protocols such as the Storage Resource Manager [13] (SRM), OGSA/DAI [14] or GridFTP [15].

Medical Data Manager project [16] (MDM) provides data annotation by centrally storing DICOM metadata from headers using the AMGA [17] metadata service, or by more complex data representations (such the BIRN Lexicon).

The Globus MEDICUS project [18] proposes the federation of DICOM medical imaging devices into healthcare Grids. It uses Globus to vertically integrate a new service for DICOM devices. This project deals with DICOM metadata from headers.

Medical data storage and processing on the GRID project [20] (MEDIGRID) is a Grid architecture which implements different applications grouped into subdomains of medical research. Users access the processing and storage resources through a web portal and developers can use regular client software to create their applications. The different basic grid middlewares supported are gLite, Globus toolkit and Unicore. As
part of MEDIGRID project introduces a Globus Security interface enhancement for the DICOM protocol, which enables secure image transfer within the grid infrastructure. A PACS component enables accessing arbitrary DICOM PACS and selected images can be transferred to Grid nodes. This component is focused on DICOM image searching and it is adapted to search within their DICOM metadata fields from headers.

The Grid projects presented above provide data annotation, data searching and data mining from DICOM metadata headers from images located in different centres. But there are also Grid projects that provide searching on DICOM-SR objects. Mammogrid [21] considers structured reports to enable context-based retrieval, but only concerning metadata from the DICOM header. The project is specific for mammograms and aims at developing a European-wide database of mammograms that will be used to investigate a set of important healthcare applications.

A more recent project is the Medical Imaging Informatics [22] (MI²), which enables DICOM medical images to be securely shared among multiple imaging centres. MI² defines grid-access-points (GAP), which are the DICOM-compliant gateways of the MI² Datagrid, managing DICOM store and query/retrieve just like a PACS. A similar gateway concept is introduced in the MEDICUS project. This project enables data annotation through DICOM-SR objects, but only using metadata from DICOM headers.

Few Grid projects allow annotation from DICOM headers and DICOM-SR content. Towards a gRid ENvironment to proCess and shAre DIc om objectS (TRENCADIS) is a middleware that implements a Service Oriented Architecture (SOA) for managing DICOM objects based on Web Services Resources Framework (WSRF), and it is implemented using Globus toolkit 4 (GT4). TRENCADIS enables sharing of DICOM objects organized into an ontological [23] and secure framework [24], using reliable and efficient transferring protocols [25]. The data are organized using the DICOM header metadata and also DICOM-SR content [26]. These annotations are kept in a flat structure, indexing data fields of DICOM-SR content in relational tables, losing the tree-like structures of the reports.

4. Grid Infrastructure

In this work we propose gLite [12] as base Grid middleware. gLite is a Grid Middleware that provides general-purpose tools for storing large amounts of data in distributed environments. Important infrastructures such as EGEE or EELA use this middleware to create biomedical applications.

gLite components that have been used in this work are the following:

User Interface (UI). The access point to the gLite is the UI. From an UI, a user can be authenticated and authorized to use the Grid resources (AMGA, LFC and SEs).

Storage Element (SE). A SE provides uniform access to data storage resources. SEs can support different data access protocols and interfaces such as GridFTP.

Local File Catalog (LFC). LFC is a catalog mapping logical references to physical file in the SE. Depending on the deployment model, LFC is installed centrally or locally. In our case, the LFC is a central catalog to all medical centers. In a LFC each file is represented by a Logical File Name (LFN). A file can be replicated at different sites, appearing as an unique LFN entry in the LFC catalog.

ARDA Metadata Grid Application (AMGA) Service. AMGA [17] provides access to metadata from DICOM headers and DICOM-SR content for files stored on the Grid, as well as a simplified general access to relational data stored in database systems.
AMGA has very good scalability and excellent performance comparable to native database access [17]. This service is deployed in all medical centre implied, and all of them synchronize the information with a central service.

The AMGA metadata service provides a layer of abstraction that works on top of the relational model. Tables are referred as collections, which have associated attributes (table columns), and are represented as directories. Table rows are called entries, and are represented as files inside the directory. A directory can also contain subdirectories, so we say that the AMGA provides a relationship between directories (a collection is within the context of another collection) that is not present in the relational model. In this article we focus on this capability of creating directory trees of database relations, as we see it a straightforward way to design a dynamic database schema that represents the structure of a DICOM-SR document content. Each entry has the default attributes of its collection and extra attribute that specifies the name of the file in the catalogue. This attribute is the entry name and should be used as the primary key of the collection. In our case the entry name is the LFN of the DICOM-SR file.

In fact, the AMGA metadata Service implements many of the functionalities of the SQL standard, but has its own command syntax.

AMGA also allows federating directories with other AMGA services, so we use the federation with proxy redirection [17] to publish the common branch to all medical centres in their DICOM-SR and launch specifics searches without data physic replication through a single access point.

5. Methodology and workflow

In this section, four data workflow processes and their methodologies are defined, which necessaries to accomplish the objectives are. Relations among gLite components involved in each workflow and the use of the developed DICOM-SR tool functionalities are outlined too.

All operations need to identify and authenticate the users, this is accomplish using x509 certificates and the security mechanism employed by gLite middleware based in Grid Security Infrastructure [18] (GSI).

5.1. Defining Templates in the VO

The workflow process of defining templates for diagnostic reports in DICOM-SR is a very complex task, especially if the objective is to define a general template to different medical centres. Except for very specific diagnostic procedures, such as CAD in mammography, (sup. 50), chest CAD (sup. 65), patient history (sup. 75), and Breast Imaging Report (sup. 79) and other emerging ones (mainly in the area of ultrasounds), there are no standardised templates. Therefore, it is reasonable that templates in different hospitals are broadly different. However, there are main parts that can be coordinated in specific, but broader, procedures, such as oncology. A clear example is the CVIMO deployment [23][26], in which five structured report templates have been developed for oncology in liver, lung and central nervous system among five hospitals.
Figure 1. Workflow to define and send DICOM-SR templates to AMGA Service.

Figure 1 step 1 shows the first task to create the templates based on DICOM-SR standard, these templates are agreed among the medical centers. Templates are described through Xml documents, defining the tree structure and the relations between the leaves that form the report and assigning the codifications according DICOM-SR standard. The contents of DICOM-SR can be seen as a data tree in which each leaf is one report element. Report elements can be of different types, each one of them conceptualized as a code (concept name) that unequivocally identifies each piece of data. Different element types have different values.

Then, the VO manager create a access point by involved centre and template in the central AMGA Server (figure 1 step 3) by executing automatically AMGA Client commands (by means of DICOM-SR tool) that create the necessaries collections, attributes and relations (figure 1 step 4).

5.2. Defining Templates in a Medical Centre

Each medical centre involved in the VO can use own-personalized templates, but providing that they include the skeleton of the general template used to create the structure of the Central AMGA. These personalized templates can introduce new leaves in the tree definition if the users of the centre consider important for their interest or internal function (these tree leaves are represented in figure 2 filled with grey colour in template 1 *). This approach keeps a degree of freedom at each centre for the definition of their own templates.

Figure 2. Workflow to define and send DICOM-SR templates to local AMGA Service.

Figure 2 step 1 shows the radiology manager of Hospital 1 defining a Xml template based on Template 1, which has a access point in Central AMGA. The Xml document is sent to local AMGA by the local administrator (figure 2 step 2) and using the DICOM-SR tool, the tree structure (figure 2 step 3) is created where local reports will be kept, leaves of the tree corresponds to an AMGA collection identified by its “concept name” and each collection stores the values and references to different DICOM-SR reports coded according to the template. Finally, another DICOM-SR tool
synchronizes, by means of federation with proxy redirection, the common leaves between Template 1* and access point of Hospital 1 in Central AMGA (figure 2 steps 4 and 5) using AMGA client commands. This synchronization provides a federated and homogeneous view of the content data locate in each common leaf when the local AMGA modifies, inserts or deletes data in these leaves.

5.3. Inserting DICOM-SR data from Medical Centres

The clinicians should send their DICOM SR-Reports to a repository if they consider that are interesting cases to share among different hospitals and in this way create a knowledge base to consult.

Diagnostic reports for a given DICOM Study can be created by following the next steps: By connecting with the PACs through a Workstation (figure 3 step 1), the radiologist visualizes DICOM images and generates an Xml document that matching with a template located in the previously selected Local AMGA (figure 3 step 2). After that, another DICOM-SR tool converts the Xml document to DICOM-SR binary file and transfers the report and its associated study to the backend (figure 3 step 3), retrieving the LFN of the report file (figure 3 step 4). If the transfer operation is performed successfully, the DICOM-SR tool gets the DICOM-SR content data and inserts it into the local AMGA (figure 3 step 5). The inserted data is accessible from Central AMGA server because of the tree has been federated previously.

5.4. Searching DICOM-SR Data

Radiologists can use DICOM-SR tool to consult DICOM-SRs content data and to download the reports and Study associated. Radiologists construct the query, basing it in a general template (figure 4 step 1), and DICOM-SR tool translates it to AMGA requests (one per hospital involved), sending them in parallel to the central AMGA
(figure 4 step 2) to corresponding access points. AMGA returns the LFNs that match with the query (figure 4 step 3). After that, DICOM-tool contacts with the backend (figure 4 step 4) and retrieves the reports and associated studies (figure 4 step 5). Finally, the radiologists can visualize the DICOM images and read DICOM-SR content data (figure 4 step 6).

6. Results

Our benchmark shows the results of an instance of AMGA 1.9 using PostgreSQL 7.4 as back-end against the results of a plain PostgreSQL 7.4 service. Both servers were executed in the same machine. We have introduced data generated by our application in the AMGA catalogue. A report structure tree has been created and filled-in with the information of 1000 DICOM-SR objects.

The plain PostgreSQL 7.4 service contains a database with two tables associated through a foreign key. In this case, table rows represent different structured reports (1000 in total). Also, table columns of each relation represent elements in the same level of the structure tree.

![Figure 5. Comparative between AMGA Service and client PostgreSQL.](image)

Information inside both servers has been randomly generated using the same probability parameters. This way, we know that equivalent queries will return approximately the same number of results in both data sets.

The clients used in the tests were the psql and the mdcli command-line tools. These programs are written in C. They were launched in the same machine as their corresponding servers so there is no network latency. Certificate authentication and SSL were disabled. Also the data returned was not processed.

The graphic shows the execution time of 1000 consecutive queries. There are no concurrent queries in the experiment. Results show that using the AMGA-based tree-like structure proposed, the introduced overhead is in the order of 20%-30%. The difference increases with the number of search keys because in our tree structure approach all DICOM-SR elements are in different collections. This means that searches will need more conditions to perform joins between tables and be equivalent to the plain relational model queries. Also, the internal database is more complex and AMGA has to perform access control checks for each entry and collection.

Anyway, this overhead will be reduced in a GRID concurrent network access scenario, [17] shows that AMGA has a better behaviour when managing concurrent client connections. Moreover, the AMGA client driver manages large amounts of retrieved data faster than the JDBC driver.
7. Conclusions and Future Works

The work introduces an approach to organise data from DICOM-SRs and to enable hierarchical and searches as fast as SQL, using parts of its metadata. It is showed that the performance of the system is reasonable for research and training activities.

The work is developing a tool that organises data integrated with a metadata catalog (AMGA), which seamlessly integrates with other gLite components as VOMS, SES, LFC and processing through the WMS and CEs.

The tool allows to structure report data into trees, this enriches the meaning of the data represented. Leaves of the tree could be organised according to the values of the container nodes, easing, from the user’s point of view, the definition of complex queries and enriching their expressivity. Moreover, the main benefits arise from enabling the federation of distributed AMGA trees from different repositories. In this way, different, but compatible structured reports can be integrated easily and securely without compromising a certain degree of freedom in each centre.

As an example for the enhanced expressivity of the use of tree structures to represent the SRs, figure 6 shows the relational and tree-like representations of a report of stratification in oncology. The key item on this report is the “Staging” field, which is computed from data of localisation, morphology and coexistence of other lesions, and drives the treatment of a patient. Figure 6 represents, on its left, a normalised relational representation of the tables PATIENTS, STAGES and LESION, which contain the data from the different patients and lesions, linked through a primary key. An SQL query to retrieve all the ID reports from the patients that have a staging value of II in which the secondary lesion is heterogeneous will look like: SELECT Patients.SRID FROM (Patients INNER JOIN Stages ON Patients.PID = Stages.PID) INNER JOIN Lesions ON (Stages.PID = Lesions.PID) AND (Stages.IDS = Lesions.IDS) WHERE ((Stages.IDS="StageII") AND (Lesions.IDL="2") AND (Lesions.CONTENT = "Heterogeneous"). Right side of figure 6 shows the same information in an AMGA tree. The AMGA query would be: Find /root/STAGE_II/Findings/Lesion_2 ‘like (CONTENT, "Heterogeneous")’. Clearly, the expressivity of the latter is much better.

Acknowledgements

The authors wish to thank the financial support received from The Spanish Ministry of Education and Science to develop the project “ngGrid - New Generation Components for the Efficient Exploitation of eScience Infrastructures”, with reference TIN2006-12890. This work has been partially supported by the Generalitat Valenciana and the Structural Funds of the European Regional Development Fund (ERDF).
References


