Volume 45, Number 1, 2004

TECHNICAL UNIVERSITY OF CLUJ-NAPOCA, ROMANIA

FACULTY OF ELECTRONICS AND TELECOMMUNICATIONS

ACTA TEHNICA NAPOCENSIS

Electronics and Telecommunications

MEDIAMIRA SCIENCE PUBLISHER

Volume 45, Number 1, 2004

TECHNICAL UNIVERSITY OF CLUJ-NAPOCA
FACULTY OF ELECTRONICS AND TELECOMMUNICATIONS

ACTA TEHNICA NAPOCENSIS

ISSN 1221 – 6542

Sponsor

S.C. 'CODEC' SRL, CLUJ-NAPOCA

IMPLEMENTATION OF A MEDICAL MULTIMEDIA DATABASE CONSIDERING ORACLE AND J2EE PLATFORM

Mircea-Florin VAIDA[1], József DOMOKOS [2]

[1] *Technical University of Cluj-Napoca ,Gh. Baritiu str. 25-27, Romania, E-mail: Mircea.Vaida@com.utcluj.ro*

[2] *Technical University of Cluj-Napoca, Dorobanţilor str. 71-73, Romania, E-mail: domi@email.ro*

Abstract: Oracle9*i* Application Server is a scalable, secure, middle-tier application server. It enables delivering Web content; host Web applications, connected to back-office applications, and access data on wireless devices. Using Oracle9*i* facilities we developed a dedicated application for medical image and multimedia content storage and retrieval considering the J2EE platform and the Oracle9i Database Server and Application Server. The application stores multimedia content such as images, audio and video data, in an Oracle9*i* database and the access to the content is guaranteed by an Oracle9*i* server application. The clients logs into system from a LAN (or Internet) having different types of security roles on the content. The application interface is presented to its users through a Web site or a file transfer client and a customer interacts with the application using a Web browser or a client application for file transfer. The application will allow information querying and searching, user management and maintenance.

Key words: Information System Management, Oracle9i Application Server, JAI, J2EE platform, Oracle interMedia, medical imaging.

I. INTRODUCTION

Medical information system management offers useful data for doctors in hospital management. The paper will present an application implemented at the Medicine and Pharmacy University of Cluj-Napoca, Radiology Department, who needs to centralize the management of patient's data like images taken from various medical equipments (CT, RMN, etc.), medical files, others content related to patient (video clips, 3D representations, sounds, etc.) and a knowledge base used for distance learning.

This application is based on the experience of an image processing and storing system built on Java platform that was developed in the Telecommunication Software Laboratory from the Technical University of Cluj-Napoca. This system consists of two main components: an image processing Java application, and a Java 2 Enterprise Edition (J2EE) server application. The image processing application uses JAI (Java Advanced Imaging) API for performing various imaging tasks on image files. The server application uses J2EE (a middle tier server solution based on Java components) for creating dynamic web content of text and images to a client represented by a web browser.

In this application by associating the users to documents it is possible to access and control the available medical information. The application provides to users through a client (web browser) the information contained into patients medical files (text, images, etc.), and also provides distance-learning information. The application contains also a Java client application, which is a J2EE client application. The client is capable to transport files between the database and MIPAS application (a high performance image visualization and processing tool), which manages DICOM files. Another client is able to manage images in other standard image formats (JPEG, TIFF, BMP, PNG, etc.), and others multimedia files (3D representations, video clips, sounds, etc.) between database and an image processing tool. The transfer of the information between client and server is done using RMI/IIOP protocol and encrypting and authentication provided by J2EE platform.

The J2EE platform uses a multi tiered distributed application model. The application logic is divided into components according to function, and the various application components that make up a J2EE application are installed on different machines depending on the tier in the multi tiered J2EE environment to which the application component belongs (Figure 1 and 2).

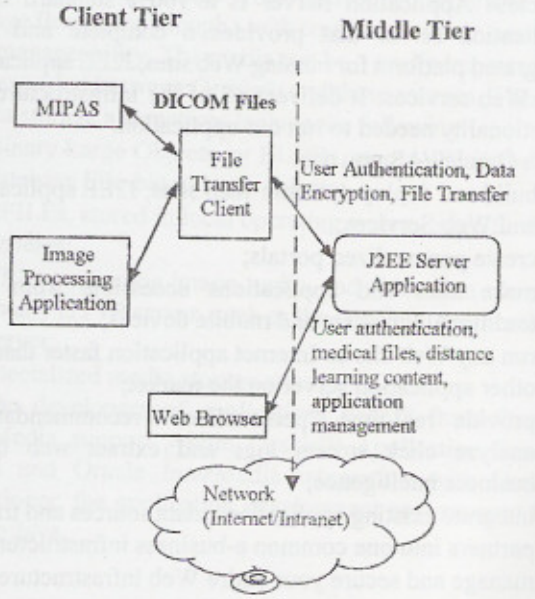

Figure 1. The client tier and middle tier, part of the J2EE multi tiered application

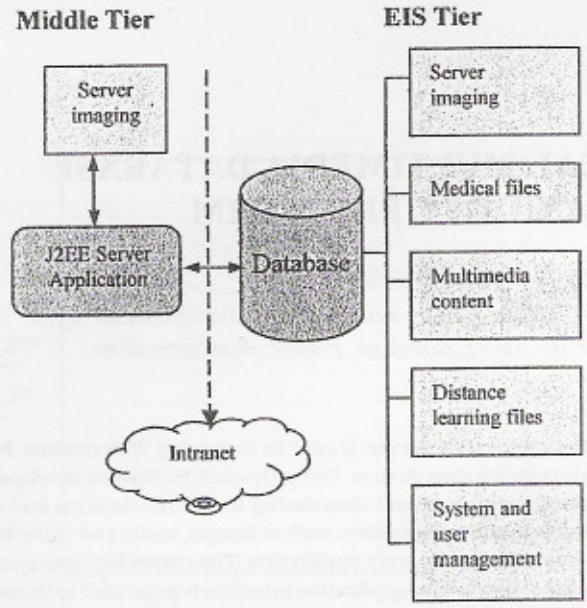

Figure 2. The middle tier and its relation with EIS Tier, also part of the J2EE multi tiered application

The J2EE application, parts shown in Figure 1 and 2 are presented in J2EE Components.

- Client-tier components run on the client machine.
- Middle-tier components (Web-tier components and business-tier components) run on the J2EE server.
- Enterprise information system (EIS)-tier software runs on the EIS server.

In our application J2EE client applications and a web browser represent the client tier. The middle tier is based on Oracle9*i* Application Server and the EIS tier is build around the Oracle9*i* Database Server.

II. ORACLE9i APPLICATION SERVER

Oracle9*i* Application Server is a 100% standard based application server that provides a complete and fully integrated platform for running Web sites, J2EE applications and Web services. It delivers all of the infrastructure and functionality needed to run our application.

With Oracle9*i*AS you can:

- build and deploy dynamic Web sites, J2EE applications and Web Services;
- create personalized portals;
- make sites and applications accessible from both traditional browsers and mobile devices;
- run any Web site or Internet application faster than any other application server on the market;
- provide real-time "personalized" recommendations; analyze click stream logs and extract web traffic business intelligence;
- integrate existing applications, data sources and trading partners into one common e-business infrastructure;
- manage and secure your entire Web infrastructure;
- scale your Web sites and applications as your business grows and delivers reliable, 24x7 service to all of your users around the globe.
- tools and programmatic interfaces to make application development easy and productive.

All of the tools needed to develop an Oracle9*i*AS are available in Oracle9*i* Developer Suite. Dynamic Web Sites Oracle9iAS provides all the capabilities you need to drive your mission critical web sites:

- Reliably Serving Web Content
- WebDAV Support
- Comprehensive XML Support
- Oracle HTTP Server

The Apache-based HTTP Server is the HTTP entry-point. It serves both static and dynamic content and provides high availability features including automatic configuration and death detection of servlets. It serves static content from the file system, and it serves dynamic content by dispatching.

2.1. ORACLE9*I*AS ARCHITECTURE

Oracle9*i*AS delivers a unique value proposition bringing you the broadest functionality offering from a single platform. Oracle9*i*AS has a layered architecture that consists of the following services:

• *Communication Services* - The entry point to Oracle9*i*AS from any client is a set of protocol handlers that manage communication for a variety of protocols including HTTP, RMI, RMI over- IIOP, SOAP, IMAP 4, Wireless Protocols (such as SMS), and Web Cache that can serve a reverse proxy servers. All of these protocol handlers are designed by deploying in a redundant, horizontal scaling model (described later in this document) with no single point of failure.

• *Application Runtime Services* - From the protocol handlers, the requests are routed to the Application Server Runtime that is a highly scalable and highly available J2EE Container. This container provides a common runtime environment for Applications developed as JSPs, Servlets, EJBs, and Web Services.

• *System Services* - The Application Server is itself designed to leverage a common set of runtime services that are necessary for J2EE Applications and Web Services - request dispatch and scheduling, resource management, resource pooling (including memory, connection pooling), clustering, fault monitoring, transaction management, and messaging).

• *Management Services* - To manage one or more instances of Oracle9*i*AS and to manage applications deployed on Oracle9*i*AS, a common set of systems management services to monitor the status, performance and faults of the system; to monitor resource consumption and usage; to manage a single instance or cluster of instances; to centrally administer security for users and applications; and to provide a comprehensive directory service framework to manage users.

• *Connectivity Services* - Applications can connect to a variety of Systems from Oracle9*i*AS including Oracle and non-Oracle Databases; Legacy Environments including CICS and EDI systems; any JCA Compliant Data source

including any Packaged Application; any Messaging System; and any Web Service.

• *Solutions* - Finally, Oracle9*i*AS does not just provide a J2EE Container and Web Services facilities but provides a comprehensive set of solutions all built on the infrastructure described above including Enterprise Portals, Enterprise Integration, Business Intelligence, Wireless, and ISV Solutions.

The figure below describes conceptually Oracle9*i*AS' architecture.

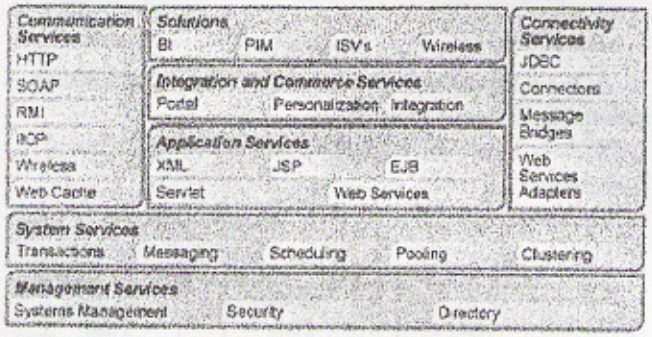

Figure 3 - Oracle9iAS Architecture

Now that we have provided an overview of Oracle9*i*AS' capabilities, we will examine each specific area in greater detail in the following sections:

- Dynamic Web Sites, J2EE Applications, and Web Services
- Enterprise Portals
- Wireless Access
- Enterprise Integration
- Business Intelligence

We follow these sections with a discussion of performance, scalability, high availability, systems management, and security.

2.2. ORACLE INTERMEDIA

Managing multimedia content for such applications presents specific issues. Large, relatively unstructured media objects come in a wide variety of complex formats, and need to be associated with the appropriate application information. Loading large volumes of multimedia content into a management system requires:

- associating the correct metadata
- indexing the content for search and retrieval
- efficiently delivering the multimedia content.

Oracle *inter*Media enable Oracle9*i* to manage multimedia content (image, audio, and video) in an integrated fashion with other enterprise information. It extends Oracle9*i* reliability, availability, and data management to multimedia content in media-rich Internet applications. As an integral part of the Oracle9*i* database server, Oracle *inter*Media data benefits from all Oracle9*i* capabilities, Oracle *inter*Media provides a set of image, audio, and video object types sufficient for most common Internet application requirements, including the popular Web formats. These object types can be extended to support many application-specific requirements, for example:

- Additional and varied formats
- New compression and decompression schemes
- Specialized indexes
- Custom query optimization and methods
- New data sources
- Specialized data processing algorithms.

Applications considering Oracle *inter*Media can easily add multiple image, audio, and video columns, or mixed columns containing any of these types as objects, to existing and new relational tables.

Oracle *inter*Media enable open, standard SQL access using native image, audio, and video data type services, operators, and metadata management. It includes Internet support for popular Web servers and authoring tools.

Oracle *inter*Media offer an array of utilities for loading multimedia content into application relational tables:

- Clipboard: A low-volume graphical utility
- API: An application programming interface
- Database table replication: Copying from one Oracle database into another
- Annotator: A sophisticated multimedia parsing utility
- SQL*Loader: A high volume, direct path bulk loader
- Oracle Internet File System.

Oracle *inter*Media extract metadata (information about the media) and associate it with the multimedia content in the media objects, at upload time. Downloading or delivery of all multimedia content in batch is supported, as streaming of certain audio and video formats.

While Oracle *inter*Media make it possible for metadata and media to be stored together within the Oracle9*i* table space, it also provides support for storage of the media outside the database making it possible to integrate archival storage. This results in considerable storage flexibility, though with some tradeoffs in security and manageability. The media can be easily imported and exported between Oracle9*i* and these sources. These media storage alternatives include the following:

- Binary Large Objects, or BLOBs, stored within the database File-based large objects;
- BFILEs, stored in local operating system- specific file systems;
- URLs containing image, audio, and video data stored on any HTTP server, such as the Oracle9i Application Server;
- Specialized media storage servers;

The developers of applications benefit from Oracle *inter*Media support within Oracle9*i* Application Server Portal and Oracle *inter*Media integration with Oracle JDeveloper, the application development environment for Oracle Java.

III. THREE TIERS ORACLE ARCHITECTURE

The Oracle *inter*Media architecture is best explained against typical three-tier architecture as depicted in Figure 4.

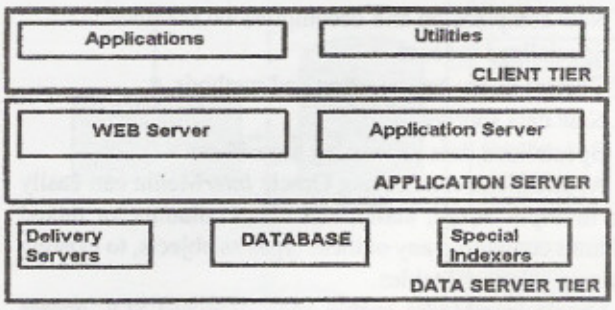

Figure 4. Three tier architecture

DATA SERVER TIER

In the first (foundation or data server) tier (Figure 5), within the Oracle*9i* database, are tables that contain *media columns*. The arrow to the right points to *externally referenced media (files)*.

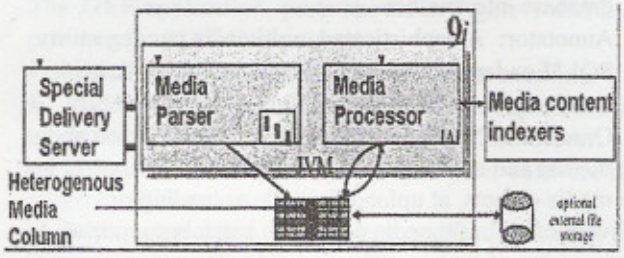

Figure 5. Data Server Tier

Above that arrow, still within the Oracle*9i* database (Figure 5), are two boxes: the *media parser* and the *media processor*. The parser takes multimedia content and parses out the format and application metadata when that content is entered into the database. The media processor, primarily based upon the Java Advanced Imaging (JAI) engine, supports the processing of images within the database. Both the parser and processor are written in Java, and run within the Oracle*9i* Java Virtual Machine (JVM).

To the left of the Oracle*9i* database (Figure 5) are the *special delivery servers*. These servers, connected by plug-ins, get the multimedia content out of the database and deliver it to thin clients (typically). The RealNetworks G2 streaming server is a good example of a special delivery server. To the right of the Oracle*9i* database (Figure 5) is a box labeled *media content indexers*. These indexers perform functions such as speech recognition and building a speech-to-text time base for the specialized multimedia content.

3.2. APPLICATION OR WEB SERVER TIER

In the middle tier (Figure 6.), is the *Web server*. It is here that Oracle *inter*Media provide access class libraries. In the case of JDBC, these are Java classes for efficient retrieval of multimedia content from the database, and delivery of it to Web browsers.

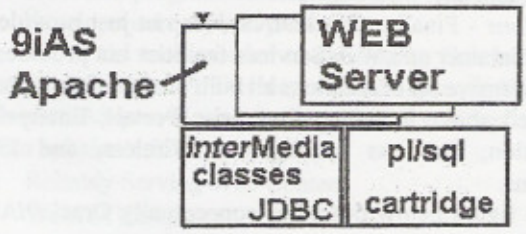

Figure 6. Middle Tier

3.3. CLIENT TIER

At the third (or top) tier (Figure 7), Oracle *inter*Media include three components:

1. *Clipboard*, two versions: NT utility and browser-based
2. *Annotator*
3. *Business Components for Java (BC4J)*

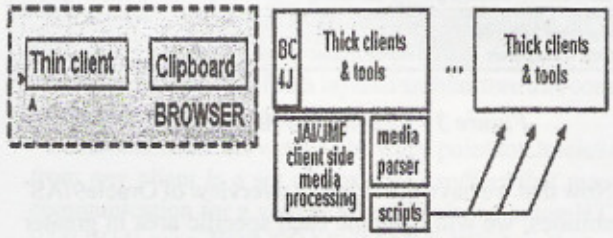

Figure 7. Client Tier

3.3.1. Clipboard: NT Utility and Browser-Based

The Clipboard component has two versions. The first version is an NT utility for browsing, uploading, and downloading multimedia content between the database and local files. The second version is browser-based and therefore portable across platforms. The second version provides a hierarchical file-oriented view of Oracle *inter*Media multimedia content.

3.3.2. Annotator

The Annotator component is a utility that parses application metadata and uploads the multimedia content and metadata into the database. Annotator is available as a Java client and as a Java bean, so it can be integrated into customer or partner applications and used in various script-based languages.

Figure 8 puts the entire architecture together and focuses on the interconnections between the three tiers.

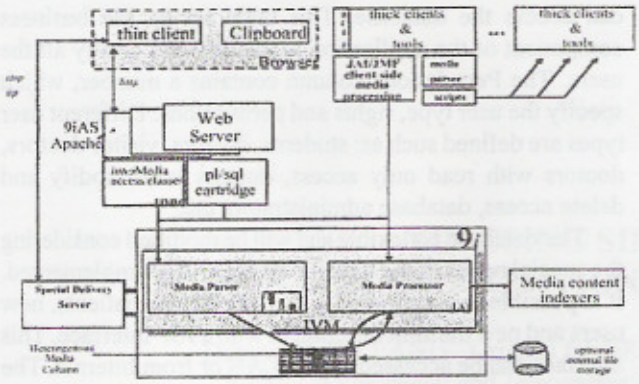

Figure 8. Complete Architecture

3.4. INTERNET INTEGRATION

Oracle *inter*Media Clipboard enables Web authoring tools to locate, retrieve, and use content from the database. During the Web page design, Oracle *inter*Media Clipboard can drag and drop multimedia content from the Oracle*9i* database onto an HTML page. It automatically creates a URL link consisting of a SQL query that points to the multimedia content in the database. When a user visits the published Web page, Oracle *inter*Media Clipboard executes the SQL query, thus retrieving the multimedia content from the database, and returns the MIME type and the multimedia content for display by the Web browser.

Oracle *inter*Media Internet facilities enable transaction control over multimedia content in Web pages. Because Oracle *inter*Media provide a late-binding feature, the most current version of the multimedia content is dynamically available whenever the Web page is visited.

Oracle *inter*Media are well suited for the emerging dynamic Web page world. *First*, it integrates management of the multimedia content with the relational decisional data used when an application chooses what content to include in a Web page. *Second*, Oracle *inter*Media provide easy-to-use, highly efficient Java classes for middle-tier query, retrieval, and output of multimedia content.

Finally, Oracle *inter*Media are already integrated with some of the leading dynamic Web page composition tools that use ASPs and JSPs, such as MacroMedia UltraDev.

IV. THE MEDICAL DEDICATED DATABASE

In the medical dedicated database we consider in this moment, 5 tables:

- Patients
- Images
- Audio
- Video
- Users

The Patients table (Figure 9), contains data (name, age, sex, contact address, phone number, e-mail address...) of all patients from the hospital and gives them a unique ID. The ID column is primary key in the Patients table and it's not possible to be null value.

The following three tables contain the multimedia data that belongs to those patients. Every recorded image, audio or video file has a patient ID attached (this shows which image belongs to which patient). As it is shown in Figures 10, 11, 12, the data types for those 3 tables are ORDImage, ORDAudio and ORDVideo *inter*Media data types.

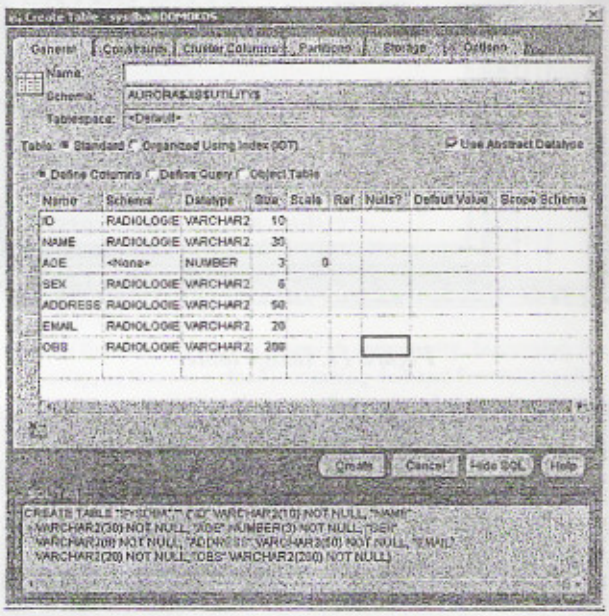

Figure 9. Patients table

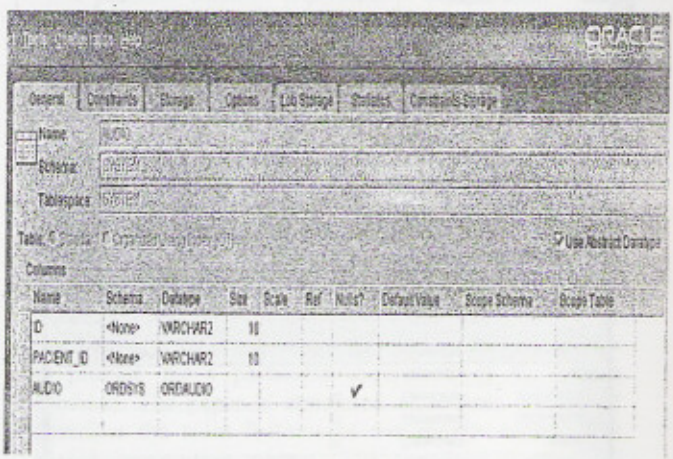

Figure 10. Audio table

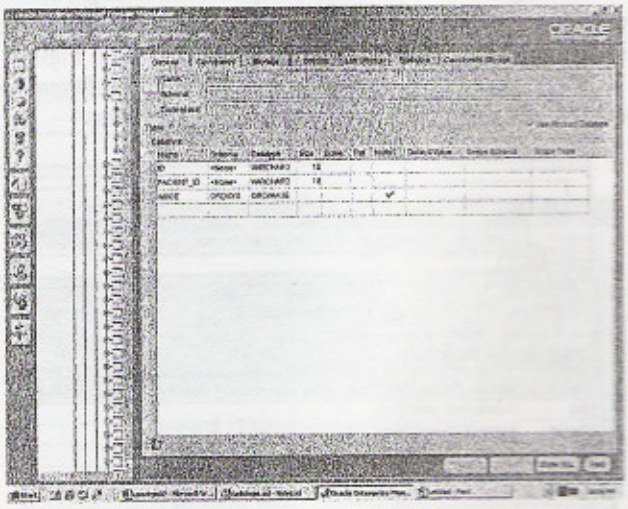

Figure 11. Image table

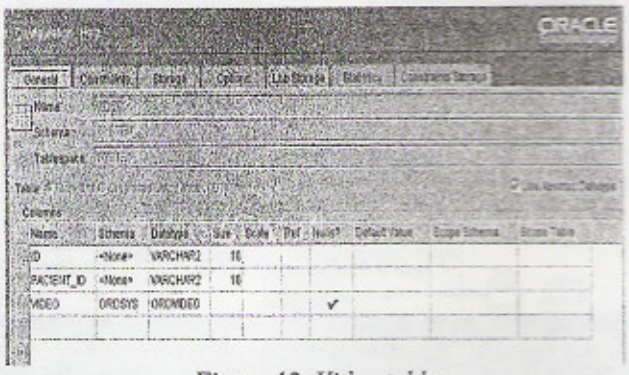

Figure 12. Video table

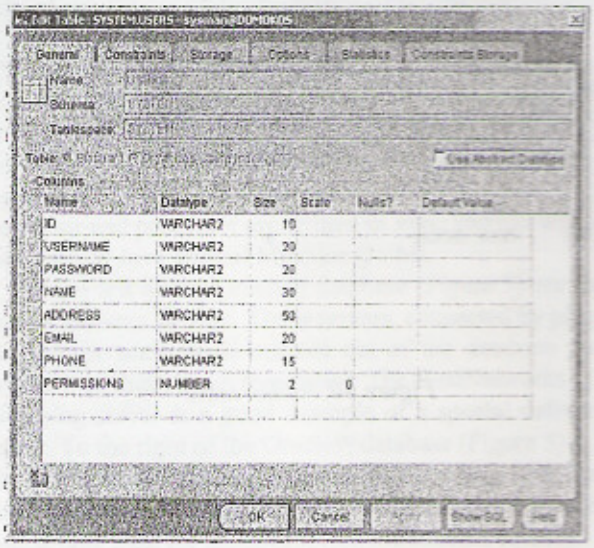

Figure 13. Users table

The last table, Users, (Figure 13), contains all the users who can access the database. This table serves the business component of the application to identify and certify all the users. The Permissions column contains a number, which specify the user type, rights and permissions. Different user types are defined such as: students, doctors, visitor doctors, doctors with read only access, doctors with modify and delete access, database administrators etc.

The database is flexible and will be modified considering the special needs of the hospital where will be implemented. It is possible in a very simple way to add new patients, new users and new multimedia content with a JSP interface. This interface can be accessed from a LAN or from Internet. The dedicated database will be implemented at the Medicine and Pharmacy University of Cluj-Napoca, Radiology Department.

V. GENERAL CONCLUSIONS

Considering multimedia database for a hospital the efficiency in the management of the hospital will increase. The advantages of this solution will be analyzed in the future after some months of using.

REFERENCES

1. The J2EE Tutorial – http:\\java.sun.com:December-2003
2. Designing Enterprise Applications with the JavaTM 2 Platform, Enterprise Edition, Nicholas Kassem and the Enterprise Team
3. Oracle9*i* Application Server An *Oracle Business White Paper*, www.oracle.com: December-2003
4. Oracle9*i* Application Server – Data Sheet, www.oracle.com: December 2003
5. Oracle8*i interMedia* Audio, Image, and Video User's Guide and Reference, Release 8.1.5
6. Mircea-F. Vaida, József Domokos, Oracle9*i In* Managing Medical Images And Multimedia Content, IEEE International Workshop, 16-18 May, 2002, UTC-N, Romania, pp. 144-151
7. Mircea-F. Vaida, Florin-O. Hurducaş, Image Processing Application Using The Java Platform And The Java Advanced Imaging API, A&QT-R 2002 (THETA 13), 2002 IEEE-TTTC International Conference on Automation, Quality and Testing, Robotics, May 23-25, 2002, Cluj-Napoca, Romania, pp. 141-146
8. Mircea-Florin Vaida, Distributed technologies and applications, technical report THR2003001, UTC-N, October 2003

CONTENTS

[continued on cover no. 3]

SUBMISSION OF PAPERS

Prospective authors are kindly requested to submit five copies of their original, unpublished contribution, which should be at most 12 pages long (8 1/2 x 11 inch sheet), double spaced (one full blank line between lines of type) using a font size of 11 points or larger. The name and the affiliation should be provided on a separate sheet.

Manuscripts must have at least 1 inch margins on all sides.

Provide an informative 5-7 keywords and 100 to 250 - word abstract, at the head of the manuscript.

References should be numbered consecutively in order of the citation in the text and listed at the end of the text.

The paper could be submitted in camera-ready form, too. Further instructions can be found at the web site http://bavaria.utcluj.ro/~atn.

Papers should be submitted to:

ACTA TEHNICA NAPOCENSIS
Electronics and Telecommunications
Faculty of Electronics and Telecommunications
RO 40027, Cluj-Napoca
Str. Baritiu Nr. 26-28
ROMANIA