

# **A Grid based Medical Image Management System using Alchemi**

**F.Maghraby**

Higher Institute of Computer and  
Information Technology,  
Elshorouk Academy,  
Elshorouk City, Egypt,

**H.M.Faheem**

Faculty of Computer and  
Information Sciences,  
AinShams University,  
Cairo, Egypt,

**M. Roushdy**

Faculty of Computer and  
Information Sciences,  
AinShams University,  
Cairo, Egypt,

## **ABSTRACT**

A large number of medical images in digital format is generated by hospitals every day. It is acknowledged that medical image databases are a key component in diagnosis. The increasing trend towards digitization of medical images creates a need of technologies for storage, and retrieval of medical images. The paper discusses how to query the medical images and presented a medical images management system based on the DICOM (Digital Imaging and Communications in Medicine) standard. Grids are a promising tool to build medical databases and to face health-related challenges involving computations over large datasets. Indeed, grids offer an infrastructure for sharing data and building virtual databases distributed over several medical sites and sharing processing power. Alchemi grid framework has been deployed to provide grid-based environment. Speeding up the retrieval and feature extraction processes was one of the major achievements of this work.

## **Keywords**

Alchemi, Database Partitioning, DICOM Standard, Grid Computing, Medical Imaging.

## **1. INTRODUCTION**

The storage and manipulation of digital images and the analysis of the information held in those images are essential requirements for next-generation medical information systems. The medical community has been exploring collaborative approaches for managing image data and exchanging knowledge and Grid technology [1] is a promising approach to enable distributed analysis across medical institutions and for developing new collaborative and cooperative approaches for image analysis without the necessity for clinicians to co-locate.

Medical diagnosis and intervention increasingly relies upon images, of which there is a growing range available to the clinician: X-ray (increasingly digital, though still overwhelmingly film-based), ultrasound, MRI, CT, PET scans etc. This reliance will increase as high bandwidth systems for picture archiving and communications are installed in large numbers of hospitals (currently, primarily in large teaching hospitals).

Digital medical images represent significant amounts of information collected by healthcare institutions about patients and their medical conditions. Clinicians need to have (strictly regulated) access to medical image data in order to conduct the analyses that can lead to detection and prevention of diseases such as breast cancer. This implies creating

diagnostic tools which allow efficient management of data, and acquisition procedures.

However, there are a number of factors that make patient management based on medical images particularly difficult. First, there is a growing range of images available to clinicians, captured by different, potentially highly variable, medical devices and where the same image may look different depending on machine and parameters. Different technical and technological solutions may result in the same medical data being stored at different locations in different formats. This leads to significant heterogeneity in the nature and content of those images. Additional heterogeneity may be due to difference in data model, data semantics and schemata, etc.

Secondly, medical images that have been captured in different geographical locations may need to be distributed across many hospitals for analysis over the lifetime of a patient (i.e. patient data are not necessarily held at a single site or accessible through a common interface). This implies a natural distribution of medical data which is particularly acute when studies are required to be conducted across sets of images resident in different computer systems.

Thirdly, very large quantities of data (from many megabytes to several gigabytes) with very complex structures are involved in medical imaging processing and exchange. In addition large numbers of exemplars are required to include sufficient numbers of abnormal cases to make significant statistical inferences in image analysis. These cases may be obtained from many geographically remote locations often crossing national boundaries. These factors lead to a need for the sharing of computation and storage resources across hospitals so that medical image data can be studied not only for individuals but also for populations of patients and at a high degree of accuracy and confidentiality as well as ease and transparency of access to the medical data. This implies providing environments where people are connected through common technologies, where new partners can join or leave without needing a major change in the system, where data can be shared inside virtual communities or organisations that respect agreed access regulations and where different fields can start working together by making resources (for example databases) accessible to each other.

Finally, the highly confidential nature of medical data requires that patient privacy and data security are key issues in collaborative medical image analysis. Data protection, ownership, data security and medical anonymity are issues that have to be addressed by any deployable distributed image analysis solution. Security in grid infrastructures may be sufficient for research, but it must be improved in the future to

ensure privacy of data. Encrypted transmission and storage is not sufficient, integrity of data and automatic pseudonymization or anonymization must be enforced to guarantee that data is complete and reliable and that privacy leakage cannot occur due to unauthorized use of the resources [1].

Computer grids are promising architectures with a strong potential for sharing resources. They are generally valued for the large computing power and data storage space they provide. Beyond this interest, grid technologies allow scientists federated in Virtual Organizations (VOs) to easily share datasets and algorithms across boundaries of their organizations. All these grid characteristics make them particularly interesting for the medical community who deals with large and fragmented amounts of medical images. As a consequence, various medical images simulation, storage, and processing applications have recently been developed on grids [2]. The problem of large scale image indexing and retrieval remains relevant for many of them.

Medical image retrieval in general has the goal to allow for retrieval of similar images or cases over very heterogeneous image collections to help the diagnostic process [3]. For large collections, this is a time consuming process, even if for single images feature extraction is often fast [4]. Parallelization of the processing is needed to keep up with the flood of images.

The proposed system uses Alchemi which is an open source software framework that can be deployed to aggregate the computing power of networked machines into a virtual supercomputer (desktop grid) and to develop applications to run on the grid. The proposed system uses the DICOM information for performing the retrieval on medical images. The retrieval is performed by extracting semantic features from the dataset values of the DICOM format. The extracted information can be used to perform the retrieval which produces a set of images relevant to the given query.

The rest of this paper is organized as follows: section 2 provides a brief introducing to grid environment and its components. Section 3 discusses Alchemi grid middleware architecture. Section 4 explains database partitioning on grid. Section 5 discusses the content based image retrieval process. Section 6 introduces DICOM standard. Section 7 explains feature extraction process over Alchemi grid middleware. Section 8 describes the computing resources and database used in the implementations process. Section 9 presents our proposed system and its modules. Section 10 discusses the experimental results and performance. Finally, section 11 provides some concluding remarks.

## **2. GRID ENVIRONMENT**

Computer grids consist of a network of computers providing distributed computing and storage resources to their users through a grid middleware. The middleware is the software layer implementing basic services to access a grid infrastructure and hiding the system complexity to the user [5].

### **2.1 Characteristics of a Computational Grid**

There are many desirable properties and features that are required by a grid to provide users with a computing environment. They are as follows:

#### **2.1.1 Heterogeneity**

The grid involves a number of resources that are varied in nature and can encompass a large geographical distance through various domains.

#### **2.1.2 Scalability**

The grid should be tolerant to handle a large number of nodes without any performance degradation.

#### **2.1.3 Adaptability or Fault Tolerant**

In a grid unexpected computational aborts, hardware or software faults etc are high. These faults are generally handled by Resource Managers.

#### **2.1.4 Security**

All the user participating computers should be protected from any malicious manipulations or interventions [6, 7].

## **2.2 Grid Components**

The major components that are necessary to form a grid. The components are as follows:

### **2.2.1 User Level**

This layer houses the Application and High level Interfaces. Applications can be varied and encompass a vast variety of problems from chemistry to Nuclear Engineering. The high level interfaces implement an interface and protocols allowing the applications and users to access the middleware services.

### **2.2.2 Middleware Level**

The major functionalities of grid systems normally occur in this layer. This layer provides many services like Resource discovery, resource scheduling and allocation, fault tolerance, security mechanisms and load balancing. It should provide the users a transparent view of the resources available.

### **2.2.3 Resource Level**

This layer typically provides local services that render computational resources like CPU cycles, storage, computers, Network infrastructure, software etc [8].

## **3. ALCHEMI ARCITECTURE**

Alchemi [9] follows the master-worker parallel programming paradigm in which a central component dispatches independent units of parallel execution to workers and manages them. This smallest unit of parallel execution is a grid thread. A grid application is defined simply as an application that is to be executed on a grid and that consists of a number of grid threads. Grid applications and grid threads are exposed to the grid application developer via the object-oriented Alchemi .NET API. Alchemi tool describes the four components which are Manager, Executor, Cross-platform manager, and Owner. These components allow Alchemi to be utilized to create different grid configurations desktop cluster grid, multi cluster grid, and cross-platform grid (global grid) [10].

### **3.1 Cluster Desktop Grid**

The cluster desktop grid (shown in figure 1) consists of a single Manager and multiple Executors. One or more Owners can execute their applications on the cluster by connecting to the Manager. Such environment is appropriate for the deployment on Local Area Networks as well as the Internet [10].

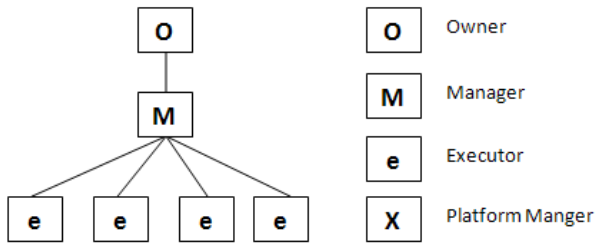


Fig 1: Cluster Deployment

### 3.2 Multi Cluster Deployment

A multi-cluster environment (shown in figure 2) is created by connecting Managers hierarchically. As in a single-cluster environment, any number of Executors and Owners can connect to a Manager at any level in the hierarchy. The key to accomplishing multi-clustering in Alchemi's architecture is the fact that a Manager at a given "intermediate" level is treated by the higher level-Manager as an Executor. Such an environment is more appropriate for the deployment over the Internet [10].

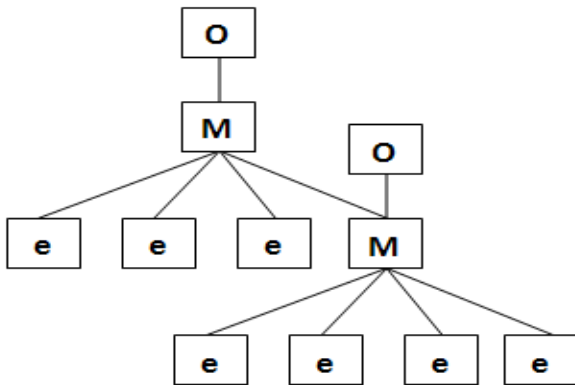


Fig 2: Multi Cluster Deployment.

### 3.3 Cross-Platform Manager

A grid middleware component such as a broker can use the Cross-Platform Manager (shown in figure 3) web service to execute cross-platform applications (jobs within tasks) on an Alchemi node (cluster or multi-cluster) as well as resources grid-enabled using other technologies such as Globus [10].

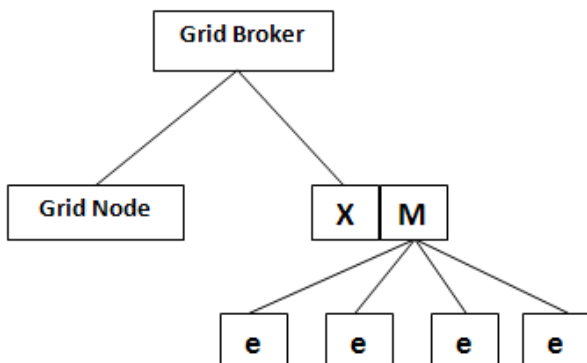


Fig 3: Cross-Platform Manager

## 4. DATABASE PARTITIONING

In this paper, the impact of executing a medical image database query application on the grid is discussed. For lowering the total computation time, the image database is

partitioned in equal subsets to be processed on different grid nodes. A theoretical model of the application computation cost and estimates of the grid execution overhead are used to efficiently partition the database. Smart partitioning of the database can lead to significant improvements in terms of total computation time [2, 11]. If the database is partitioned in bags of images to be analyzed, each bag can be analyzed by a single computing job. If one bag is representing one image so that all images could be processed in parallel, then Alchemi threads could work in parallel. Hence, the execution time would be the maximum of the execution times of each image processing.

## 5. CONTENT BASED IMAGE RETRIEVAL

A large number of medical images in digital format are generated by hospitals and clinics every day. Such images constitute an important source of anatomical and functional information for diagnosis of diseases, medical research, and education. It is well known that medical image databases are the key component in diagnosis and preventive medicine. This increasing trend towards digitization of medical images creates a need of technologies for storage, organization, and retrieval of the medical images. Content based image retrieval (CBIR) is the digital image searching problem in large databases that makes use of the contents of the images themselves rather than relying on the textual information [12]. These techniques use the automatically derived features (low level feature) such as color, texture and shape as search criteria. Medical images generated in hospitals contain semantic information (high level feature). This information can be used to retrieve the images [13, 14].

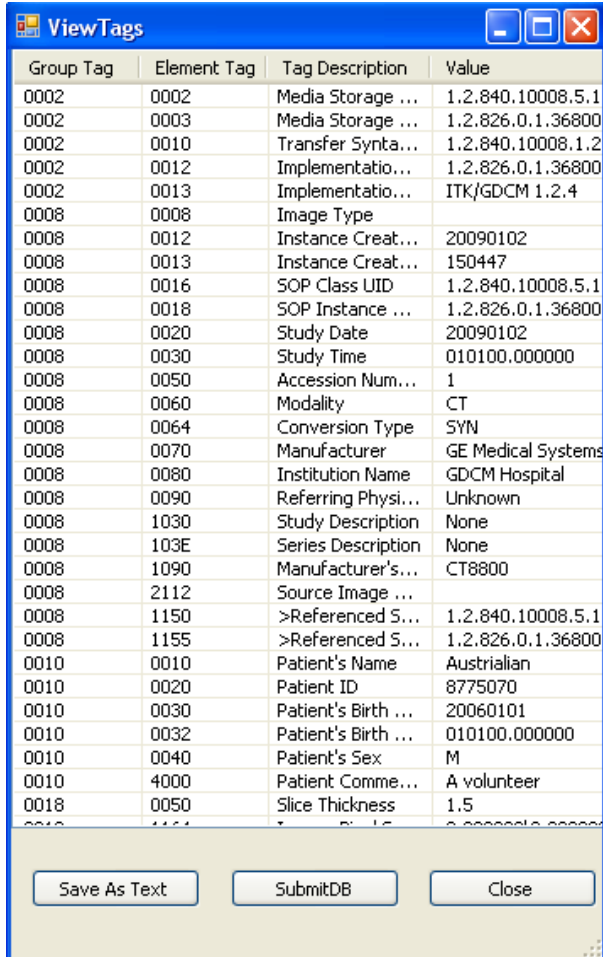
## 6. DICOM STANDARDS

DICOM (Digital Imaging and Communications in Medicine) [15] is a standard for medical image data communication and storage. The DICOM was created by the National Electrical Manufacturers Association (NEMA) to aid the distribution and viewing of medical images, such as CT (Computed Tomography) scans, MR (Magnetic Resonance), and US (Ultrasound). Imaging equipment used in hospitals generates images which are in DICOM format. It is a standard format used to obtain, store and distribute medical images. DICOM comprise standardized textual descriptions of study, patient, body region examined and modality. A single DICOM file contains both a header (which stores information about the patient's name, the type of scan, image dimensions, etc), as well as all of the image data. This is different from the popular Analyze format, which stores the image data in one file (\*.img) and the header data in another file (\*.hdr). The DICOM header size varies depending on how much header information is stored. The header describes the image dimensions and retains other text information about the scan. DICOM files are composed by one image and tags describing the image. Tags are textual or numerical sequences of <attribute, value> pairs. The textual information is similar to the semantic information. For all the DICOM files the image and the relevant tags are extracted and are stored in the database. The image is stored in jpeg file format. The extracted semantic information is stored in the database which is used during the retrieval process [13, 16].

## 7. FEATURE EXTRACTION OVER ALCHEMI

Medical Image contains the information of the image itself and includes a lot of the corresponding information, e.g. patient information, equipment information, image pixel

information and so on. DICOM standard is an object instance (images and related information) of the real world that as a DICOM file. Such a DICOM file contains a series of data sets. The data set is composed by images and many of the "logical group" of the images information, Such as group number "0010" data set stands for the corresponding information of patients.



Group Tag	Element Tag	Tag Description	Value
0002	0002	Media Storage ...	1.2.840.10008.5.1
0002	0003	Media Storage ...	1.2.826.0.1.36800
0002	0010	Transfer Synta...	1.2.840.10008.1.2
0002	0012	Implementatio...	1.2.826.0.1.36800
0002	0013	Implementatio...	ITK/GDCM 1.2.4
0008	0008	Image Type	
0008	0012	Instance Creat...	20090102
0008	0013	Instance Creat...	150447
0008	0016	SOP Class UID	1.2.840.10008.5.1
0008	0018	SOP Instance ...	1.2.826.0.1.36800
0008	0020	Study Date	20090102
0008	0030	Study Time	010100.000000
0008	0050	Accession Num...	1
0008	0060	Modality	CT
0008	0064	Conversion Type	SYN
0008	0070	Manufacturer	GE Medical Systems
0008	0080	Institution Name	GDCM Hospital
0008	0090	Referring Physi...	Unknown
0008	1030	Study Description	None
0008	103E	Series Description	None
0008	1090	Manufacturer's...	CT8800
0008	2112	Source Image ...	
0008	1150	>Referenced S...	1.2.840.10008.5.1
0008	1155	>Referenced S...	1.2.826.0.1.36800
0010	0010	Patient's Name	Austrianian
0010	0020	Patient ID	8775070
0010	0030	Patient's Birth ...	20060101
0010	0032	Patient's Birth ...	010100.000000
0010	0040	Patient's Sex	M
0010	4000	Patient's Comme...	A volunteer
0018	0050	Slice Thickness	1.5

Buttons at the bottom: Save As Text, SubmitDB, Close

**Fig 4: DICOM Tags**

A series of algorithms were implemented in order to extract alphanumeric information and image(s) from the DICOM standard files. The extracted data are stored in a database with a specific structure and can be visualized and subjected to some processing. The DICOM files contain both alphanumeric information (the name of the patient, date of birth, diagnosis, the name of the doctor) and one or more images compressed or in raw format. These files cannot be viewed on a computer. In order to do that, the DICOM files must be processed; the information must be extracted and eventually stored in a database. So, the information can be viewed anytime, subjected to some processing or queries. After extracting all needed features from images, it will be

stored in SQL database server to be used in searching process, and then the database is transported to computation nodes and partitioned to equally sized subsets [17, 18]. Some extracted DICOM information can be observed in figure 4.

Extracting features on one image takes at least a couple of seconds [19]. So, it can take hours for large image database (about 56 hours for a collection of 100'000 images). To speed up the performance, the feature extraction task can be parallelized in such a way that some of the tasks can be executed over different nodes .Alchemi Grid was used to parallelize the execution of feature extractions and indexing processes.

## 8. METHODS

In this section the setup for the development is explained including computing resources and database used.

### 8.1 Computing Resources Available

The proposed system was tested on a cluster in AinShams University. The cluster consisting of 2 blade chassis; each has 6 blade server (quad core processor, 2.8 GHZ, 24 GB RAM, 3\*300 GB hard disk) in addition the 4TB storage server. Typical environment is implemented such that ESXi is deployed and a set of VMWare (Virtual Machine Ware) are running on window operating system using them as computing nodes.

The system was tested on 6 blade server (one blade chassis). Alchemi desktop grid is constructed by deploying a Manager node which also works as executor, and deploying 5 Executor nodes .The six executors configured to connect the Manager. One or more Users can execute their applications on the cluster by connecting to the Manager.

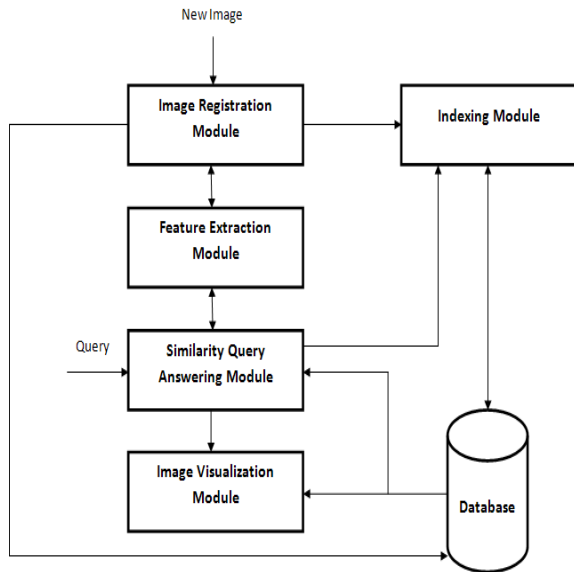
### 8.2 Dataset Used

The system was tested on set of 100,000 images with 40 Million feature records were extracted (over 16 GB). The features of every image can be calculated in parallel and independently of other images. The images are grouped in blocks of 1000 to be computed on the same node.

## 9. PROPOSED SYSTEM ARCHITECTURE

The general architecture of the system includes five modules: (a) Image Registration Module, (b) Feature Extraction Module, (c) Indexing Module, (d) Similarity Query Answering, and (e) Image Visualization.

New images are registered to the database, through the DICOM handler. This module stores every new image in the database, linking it to the corresponding features data. For each new image, the image registration module makes a request for the feature extraction module to obtain the image feature vector. The feature vector is then sent to the indexing module, which links the features with the image stored in the database, and includes the features in the indexing structure. When the images are already stored, the system can answer similarity queries, which are posed through the similarity query answer module.



**Fig 5: Proposed Architecture**

Whenever a similarity query is received, this module calls the feature extraction module to obtain the feature vector of the query image, and requests the indexing module to find the images in the database that are the answer for the posed query. It then calls the visualizer to display the resulting images. The visualizer can also prepare the attributes of images, so the image and its attributes can be submitted to the DICOM handler, to be distributed to every existing workstation already prepared to support this kind of data .the proposed architecture is shown in figure 5.

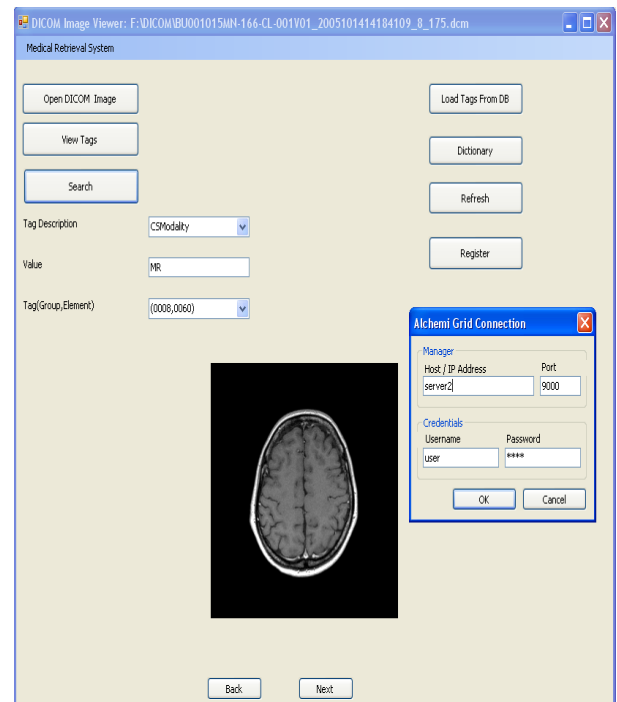
## 10. EXPERIMENTAL RESULTS AND DISCUSSION

A dictionary was constructed containing all DICOM tags to allow users to choose tags related to their query attributes. Assume that the user query is to retrieve all the images of US modality for Patient Name “Smith”. In this example the user must specify Modality, Patient name attributes through the user interface. The User Interface is shown in Figure 6.

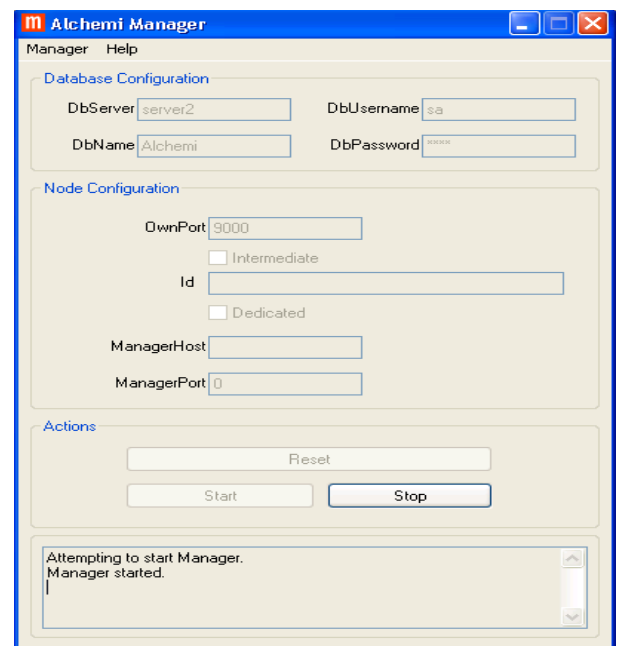
When the user selects the search operation, the system connects to grid middleware, and then Alchemi imitates traditional multi threaded programming. GThread is a grid thread and GApplication is application thread. “Just in time scheduler algorithm” is used for splitting the comparison job into different threads. Assume that T1, T2, T3....Tn are the threads generated by Alchemi. Alchemi owner provides an interface with the grid application between the application developer and the grid. The owner submits the completed threads to the Alchemi Manager. The Alchemi Manager manages the execution of threads responsible for the searching process in its own dataset.

The executors register themselves with the Manager which in turn keeps track of their availability. Threads received from the Owner are placed in a pool and scheduled to be executed on the various available Executors. The Executor accepts threads from the Manager and executes them. Executor API is used to make an interface with the Alchemi Manager. Figure 7 shows the configuration of Alchemi Manager. In the figure, setup connection enables or starts the Alchemi manager. Figure 8 shows the configuration of Alchemi Executor that

contains the host and the port number of connectivity, and credentials required to login on to the system. Figure 9 shows Console Form .This form provides the system statistics and real time graph of power availability and usage.



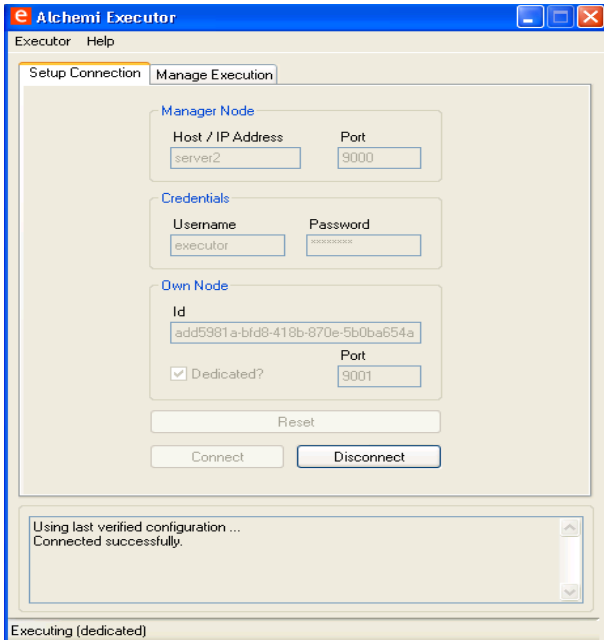
**Fig 6: User Interface**



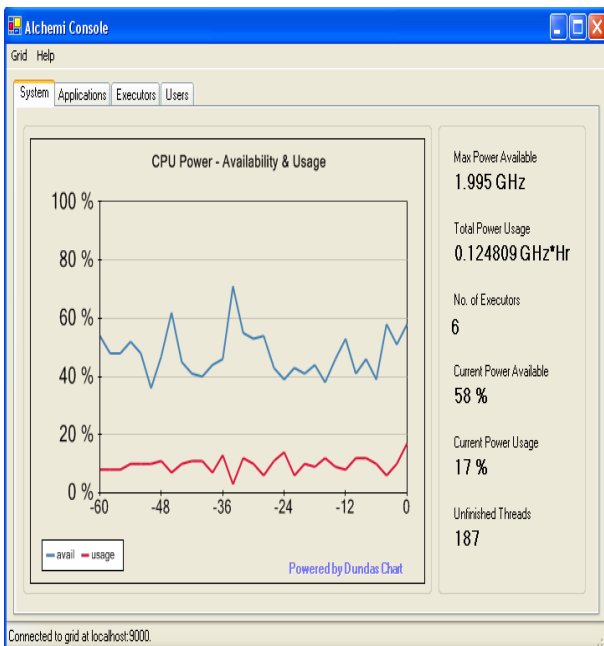
**Fig 7: Manager Window**

Figure 10 shows a plot between number of images and total time (in seconds taken by the all threads to complete execution) with varying numbers of Executors enabled. The figure shows that increasing the number of executor can be a sufficient way to handle large datasets. The gap in runtime greatly increases as the dataset grows larger in size. The results show that partitioning the dataset on different executor nodes while there is no task dependency can lead to a

significant improvement in the retrieval time. These results demonstrate that the grid computing can dramatically reduce the time required to retrieve medical images. Alchemi grid was used to parallelize the feature extraction process. For a set of image consisting of 1000 images, the feature extraction algorithm can take an average time 3.5 minutes using Alchemi.

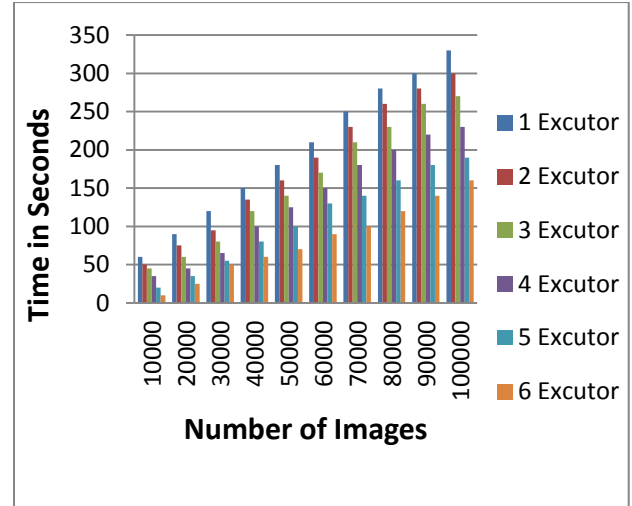


**Fig 8: Executor Window**



**Fig 9: Console Form**

Consequently 100 set of images have taken 350 minutes on the grid. As mention above the feature extraction and indexing of large image database can take 56 hours for a collection of 100'000 images so, the proposed system provides feature extraction and indexing processes at speeds up of 10 times faster.



**Fig 10: A Plot of image size versus execution time on a standalone Alchemi cluster with varying numbers of executors enabled.**

## 11. CONCLUSION

In this paper, we propose a grid based system for retrieving medical images. Grids are promising architectures that can bring different solutions to medical image storage and retrieval problems. Computational solutions such as Grids are necessary if data is to be treated or analyzed to improve the care processes and provide diagnosis aid. Currently the clear tendency for high performance computing is towards parallel processing using standard components. The availability of multicore CPUs is also accelerating this process. To speedup retrieval process, the database was partitioned into equally sized subsets, so the queries can benefit from the grid computing parallelism and execute different queries, each on a different subset of images. Features extraction on large number of images takes hours, the grid technology was to accelerate this process. We can see that Alchemi framework is ideally qualified to be deployed in medical image retrieval system .Of course, as the number of executors increases, the overall system performance increases. The system is considered a step towards a complete grid-based implementation for a complete medical management system.

## 12. REFERENCES

- [1] Rogulin, D., Estrella, F., Hauer, T., McClatchey, R., and Solomonides . T.2004.A Grid Information Infrastructure for Medical Image Analysis. In Proceedings of DiDaMIC Workshop (MICCAI'2004).
- [2] Montagnat, J. and al, e.2004. Medical image content-based queries using the Grid. Proceedings of the first European HealthGrid conference .pp.142-151. Lyon, France.
- [3] Müller ,H., Michoux ,N., Bandon ,D., and Geissbuhler ,A.2004. A Review of Content Based Image Retrieval Systems in Medicine - Clinical Benefits and Future Directions. International Journal of Medical Informatics, 73(1), 1-23.
- [4] Zhou, X. , Pitkanen ,M., Depeursinge,A., and M'uller, H. 2008. A Medical Image Retrieval Application Using Grid Technologies To Speed Up Feature Extraction. In ICT4Health, Manila, Philippines.
- [5] Camarasu, S., Benoit-Cattin, H., Montagnat, J., and Racocanu, D. 2008.Content-Based Medical Image



- Indexing and Retrieval on Grids. First International Symposium on ICT for Health, Ateneo de Manila University, Philippine J Info Tech. Manila, Philippines.
- [6] Baker, M.A., Buyya,R., and Buyya,D.2000. The Grid:International Efforts in Global Computing. International Conference on Advances in Infrastructure for Electronic Business, Science, and Education on the Internet ,SSGRR 2000. Rome, Italy.
- [7] Germain, C. , Néri,V., Fedak,G. and Cappello,F.2000.XtremWeb : Building an Experimental Platform for Global Computing. First IEEE/ACM International Workshop, GRID2000 .pp. 91-101, Bangalore, India.
- [8] Arnold,D. C., Vadhiyar, S. S. , and Dongarra, J. 2001.On the Convergence of Computational and Data Grids. Parallel Processing Letters, 11(3), 187-202.
- [9] Luther ,A., Buyya,R., Ranjan ,R. and Venugopal,S. 2003 . Alchemi : A .NET-based Grid Computing Framework and its Integration into Global Grids .Technical Report, GRIDS-TR-2003-8, Grid Computing and Distributed Systems Laboratory, University of Melbourne, Australia.
- [10] Dhivya, M., and Ruba, K. 2012. Building Grid based Application for the Management of Medical Image Data using Alchemi . International Conference on Recent Trends in Computational Methods , Communication and Controls (ICON3C) , Proceedings published in International Journal of Computer Applications® (IJCA), Foundation of Computer Science, New York, USA.
- [11] Montagnat,J., Breton ,V., and Magnin ,I. E. 2005. Partitioning Medical Image Databases for Content-Based Queries on a Grid. International Journal of Supercomputer Applications, 44(2),154-160.
- [12] Caicedo,J.C. , Gonzalez,. A., and Romero ,E. 2008.A Semantic Content-Based Retrieval Method for Histopathology Images. In H. Li (Eds.), Information Retrieval Technology: 4th Asia Information Retrieval Symposium ,AIRS 08, LNCS 4993, Springer-Verlag, Berlin, Heidelberg .pp. 51–60.Germany.
- [13] Selvarani, A.G., and Annadurai, S. 2007.Medical Image Retrieval by Combining Low Level Features and DICOM Features. International Conference on Computational Intelligence and Multimedia Applications, Vol.1, IEEE .pp. 587 – 591. Sivakasi, Tamil Nadu,India.
- [14] Weyand,T., and Deselaers , T. 2005 . Combining Content-based Image Retrieval with Textual Information Retrieval. RWTH Aachen Department of Computer Science, RWTH Aachen University. Aachen, Germany.
- [15] Oleg S. P. 2012. Digital Imaging and Communications in Medicine (DICOM).Heidelberger Platz 3, Berlin: Springer-Verlag.
- [16] Stanescu ,L., Burdescu, D., Ion, A., Caldare ,A., and Georgescu ,E. 2006. Extracting, Storing and Viewing the Data from DICOM Files. Med-e-Tel: International eHealth, Telemedicine and Health ICT Forum for Education, Networking and Business.
- [17] Lu, X., Zhang, M., Yang, L., Zhao, Y., and Liu, J. 2012.Research and Implementation of Medical Images Management System Based on DICOM Standard. International Conference on Biological and Biomedical Sciences Advances in Biomedical Engineering, Vol.9, .pp.140-146. Shanghai,China.
- [18] Traina, A. J. M., Rosa, N.A., and Jr, C.T. 2003. Integrating Images to Patient Electronic Medical Records through Content-Based Retrieval Techniques. In proceeding of 16th IEEE Symposium on Computer-Based Medical Systems (CBMS) .pp. 163-168, NY: IEEE Computer Society.
- [19] Yuanqing ,L., Lv ,F., Zhu,S., Yang,M., Cour ,T. and Yu ,K. 2011.Large-scale Image Classification:Fast Feature Extraction and SVM Training. Computer Vision and Pattern Recognition (CVPR), IEEE Computer Society .pp.1689 – 1696. Providence, RI,USA.