Available online at www.elixirpublishers.com (Elixir International Journal)

Computer Science and Engineering

Elixir Comp. Sci. & Engg. 45 (2012) 7709-7713

ISSN: 2229-712X

Health Care System with Audio and Video Facilities

Narendra Kohli and Nishchal K. Verma

Electrical Engineering Department Indian Institute of Technology Kanpur, India.

ARTICLE INFO

Article history:

Received: 17 January 2012;

Received in revised form:

17 March 2012;

Accepted: 2 April 2012;

Keywords

Red5,
Health Care System,
Videoconferencing,
Patient record,
SQL Server.

ABSTRACT

In this paper, Smart Card based Online Health Care System with low cost Audio and Video Facilities have been discussed. For providing a good quality of treatment to the patients in the hospital, it is required to automate and integrate the hospitals of country via internet. After registration, patient will get a smart card with 10 digits unique registration number/ PatientID with personal information in the hospital. All the doctor prescriptions, test reports (blood, urine etc), images (MRI, CT-Scan, ECG etc.) will be stored in the databases of the hospital server (local server) as per the PatientID. All the local servers of hospitals are connected with one centralized server and time to time all the patient data from the local hospital server will be transferred to main centralized server. Patient information, doctor information and hospital information etc. can be retrieved from any hospital of any patient as per the PatientID. There may be a situation where patient is present in a rural hospital where sufficient expert doctors are not available. In such situations, doctor present in the remote hospital may need to contact an expert to take his advice for better treatment of the patient. In this work, a Graphical User Interface for low cost Videoconferencing System for Online Health Care System is also discussed, specifically to support consultations among remotely placed Patients, rural health workers and specialists in the urban cities and provide a secure access to remote patient records. A low-cost web-based Videoconferencing System using Open Source Technologies such as Red5 etc. has been developed. We have used live video/audio, text chat and video recording/archiving tools and features for the same. Online Health Care System application has been developed as front end in DOT NET [4] and back end in SQL Server. SQL Server tuning & other features are implemented for fast retrieval of patient data while designing the application of Health Care System.

I. Introduction

A complete network of Health Care providers in a specified geographic area is known as Health Care System. Health Care System is an organized society. It connects people with the hospitals, doctors and other health related information of the patients. Health Care System increases the accuracy and speed of patient information circulation which results in better services for the patients. It also increases the working efficiency of the hospital system [1, 2, 3].

The proposed web based application has been developed to obtain, store, analyze, process and use the patient information that concerns with the doctors, hospitals, laboratory tests etc. This application can serve large number of hospitals and provide an interface for their interaction. As per the application, the developed system generates ten digits unique Registration Number/ PatientID (three digits for hospital identification number and remaining seven digits for patient identification number) for registration purpose in a hospital. The smart card is issued to the patient containing PatientID with other basic information like name, address, phone number etc. Patient information like doctor diagnosis, test reports, MRI and CT-scan images etc. are stored in the databases of the hospital server as per the PatientID. For future usage, it helps in quick diagnosing the illness of the concerned patient. While visiting to the hospital for treatment, patient will need to carry only smart card and the administrator of the hospital or the doctor can use this smart card through card reader to extract the patient related needed information. To increase the performance of the application, we have covered all kinds of optimizations at database design and application design levels. We have also carried out the experimentation to achieve proper memory utilization and optimizations of queries for accessing the database. We have removed all kinds of redundancies using normalizing the database.

This paper is organized into total VII sections. A brief discussion of the Database and Application Design Aspects system is presented in Section II. In section III, we have discussed Database Block Size Optimization. Section IV discusses about the case study. Section V discusses about the Graphical User Interface for Videoconferencing and Text Chatting using Open Source Technologies. Working Snap shots of Videoconferencing using Red5 and Text Chatting have been discussed in section VI. Finally, the conclusions and future works are drawn in Section VII.

II. Database and Application Design Aspects

This application can serve large number of hospitals and provide an interface for their interaction. As per the application, the system will generate 10 digits unique registration number/ PatientID (three digits for hospital identification number and remaining seven digits for patient identification number) for every patient. According to the PatientID, all the patient information can be retrieved from the databases of local hospital server or from the main or centralized server of the hospital

Tele:
E-mail addresses: kohli.hbti@gmail.com

network. All patient related information will be on a centralized database server, so the movement of files is not required as the case in distributed database server. All the files, laboratory test reports, medical images, doctor prescriptions, scan documents and diagnosis reports of the patients can be retrieved from the local/centralized database server. Accessing the information related to patients has been controlled at various levels of access control and it is preventing the unauthorized access from the database. All possible redundancies have been removed using normalization of the databases. We have carried out all types of experimentations to achieve proper memory utilization and optimization of queries for accessing the database. To increase the performance of the application, we are considering the following database and application design aspects.

1. All the tables have been normalized.
2. Maximum utilization of memory.
3. Necessary constraints have been imposed on tables.
4. Selecting best possible structure for queries.
5. Tuning and optimizing queries.
6. Optimal selection of join order and join algorithm.
7. Managing insert operation to reduce load from server.
8. Front end should be strong enough to reduce basic user faults and provide easy interaction.
9. Speed up the insert, update and delete operations.

Some optimization steps are as follows

1. As per the requirements of the application, respective tables are created to store that information. Databases are normalized up to Boyce Codd Normal Form (BCNF) level and 14 tables are created.
2. Size of attributes (their respective data types) is further minimized in order to compact the record size, so that maximum number of records can be stored in a single data page. Selected queries are designed to handle all possible search criteria.
3. Order of attributes in composite indexes has been used.
4. Due to monotonically increasing nature of PatientID, index portioning and reverse key indexing techniques are proposed.
5. In a multi-table join queries, we have selected the appropriate order of the tables.
6. Proper join algorithms and hints are used for the optimizer.
7. To improve the insert, update and delete operations by the hospital, we are storing the patient data at the hospital local server. It is reducing the load of main centralized server and problem of unbalancing of index structure.
8. Every day at a particular time, all patient related informations of all the local servers of the hospitals are automatically transferred to the main centralized server.
9. Transaction log is permanently off at centralized server as we are not requiring any log for recovery because all the information are at hospital local server and can be reproduced easily.
10. All primary key, foreign key constraints have been removed from the centralized server as those constraints are already checked at local server of the hospitals. So there is no need to revalidate the already validated information. It has improved the performance of data loading at the centralized server.
11. For updating centralized server, we have recommended to use bulk copy and bulk insert schemes.

III. Database Block Size Optimization

Database size has been optimized for efficient utilization of memory i.e. uses of smaller data types in database tables. Database software defines some fixed size storage structure to store table records and other information. These structures hold some bits for system information and rest for data. The ratio between system information and table records in a storage structure plays an important role as it defines the memory organization of database. If portion of these structures that holds table records is used properly then memory will be utilized efficiently and definitely increases the performance of the system / application.

SQL Server stores data in a special structure called index page that is 16 kilobytes (16384 bytes) in size. 152 bits are used to store system information and 16232 bits are used to store user's data. So, when the table's row size is 8126 bits, then only one row will be placed in each data page and 8106 bits of memory are unusable. But, if the table row size is 8116 bits then two rows can be stored within a single page. We have designed our tables in such a way as to fit maximum number of rows into one data page. To fit maximum number of rows into one data page, we have specified the narrowest columns to fast read and write of data. For maximum utilization of memory and speed up read / write operations of the databases, we have proposed that the row size of a table should be some devisor of 16232 bits, when the page size is 16 kilobytes or divisor of 8040 bits when the size is 8 kilobytes or divisor of 3944 bits when the page size is 4 kilobytes etc. Figures 1and 2 show the partial and full utilizations of memory.

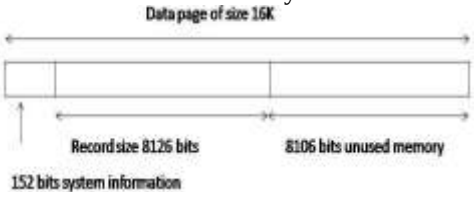


Figure 1: Poor utilization of Memory

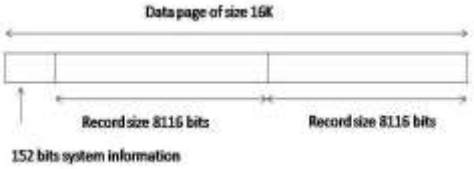


Figure 2: Full Utilization of Memory

In SQL Server, we can also compress data page size to 1 kilobytes, 2 kilobytes, 4 kilobytes or 8 kilobytes. If the size of index page is 16 kilobytes then table size can be considered in one of the following magic numbers for efficient memory utilization [14,15].

1 2 4 8 2029 4058 8116 16232 bytes

For 8 kilobytes of data page size, the magic numbers are

1 2 3 4 5 6 8 10 12 15 20 24 30 40 60 67 120 134 201 268 335 402 536 670 804 1005 1340 1608 2010 2680 4020 8040 bytes.

For 4 kilobytes data page size, the magic numbers are

1 2 4 8 17 29 34 58 68 116 136 232 493 986 1972 3944 bytes.

For 2 kilobytes of data page size, the magic numbers are

1 2 3 4 6 8 12 24 79 158 237 316 474 632 948 1896 bytes.

For 1 kilobytes of data page size, the magic numbers are

1 2 4 8 109 218 436 872 bytes

IV. Case Study

To identify the bottleneck and performance related issues, we have done different case studies using different methods. Thereafter, we have analyzed the queries and their execution paths. For simulation of the work, we have taken approximate seven lac records in every table of the databases. We have used the following query for the analysis. This query selects patient details from four main tables by applying inner joins.

Query:

```
SELECT Patient1.PatName, Patient1.Address, Patient1.ContactNo, Patient1.Sex, Patient1.Occupation, Patient1.Guardian,
```

```
Patient1.RegNo, Patient1.ReferredBy, Patient1.BloodGroup,
Patient1.PatientHistory, Patient1.Disease,
Patient1.DiseaseCatageory,
Patient1.EntryDate, PatientRoomIPD.HospitalID,
PatientRoomIPD.RoomID, Patient- RoomIPD.StartDate,
PatientRoomIPD.EndDate, PatientRoomIPD.Status,
PatientRoomIPD.TotalCharge, PatientDoctor.DoctorID,
PatientDoctor.DoctorName, PatientDoctor.StartDate AS
TreatmentStartDate,
PatientDoctor.EndDate AS TreatmentEndDate,
PatientDoctor.No Of Visits, PatientDoctor.Charges AS
TreatmentCharge, PatientLab.LabNo,
PatientLab.TestNo, PatientLab.TestDate, PatientLab.Unit
FROM Patient1
INNER JOIN
PatientRoomIPD ON Patient1.PatientID =
PatientRoomIPD.PatientID
INNER JOIN
PatientDoctor ON Patient1.PatientID = PatientDoctor.PatientID
INNER JOIN
PatientLab ON Patient1.PatientID = PatientLab.PatientID
where Patient1.PatientID=ID;
```

We have executed above query by putting it into stored procedure using query-analyser. We have used random PatientID several times on hardware configuration: Intel Duel Core 2.8GHz 2GB RAM as per Table 1.

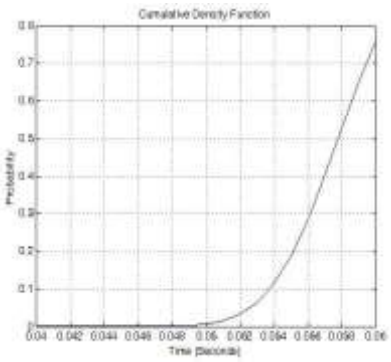
Case 1: Retrieval Time with Traditional SQL Server Database:

Retrieval mean time of PatientID with traditional SQL server database in stored procedure on query analyzer is 0.0577 second.

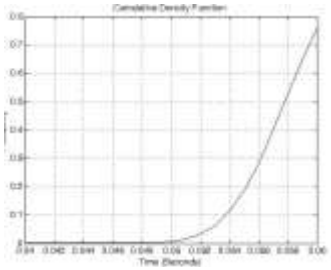
Cumulative Density Function and Probability Density Function:

The Probability Density Function (PDF) of a continuous random variable is a function which can be integrated to obtain the probability that the random variable takes a value in a given interval. PDF is used to find the point of normal distribution curve.

Cumulative density function (CDF), or just distribution function, describes the probability that a real-valued random variable X with a given probability distribution will be found at a value less than or equal to X. So, it is the area so far function of the probability distribution. Cumulative density functions are also used to specify the distribution of multivariate random variables. PDF and CDF plots shown in Figure 1 are obtained using the values of the retrieval times on query analyzer given in Table 1.



(a)



(b)

Figure 1: (a) Probability Density Function, (b) Cumulative Density Function

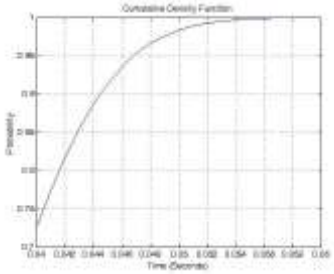
Case 2: Retrieval Time with Optimized SQL Server Database

We have executed above mentioned query with new databases (after block size optimization) by putting it into stored procedure using query analyzer. We have taken random PatientID eight times. The results are as per Table 2.

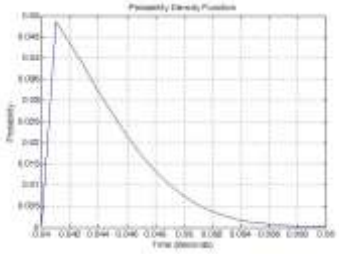
Comparison of the retrieval mean times of the traditional and optimized databases in stored procedure on query analyzer is given below. We can see an improvement in case of optimized database.

1. With traditional database, retrieval mean time is 0.0577 second.
2. With optimized database, retrieval mean time is 0.0361 second.

PDF and CDF plots shown in Figure 2 are obtained using the values of the retrieval times on query analyzer given in Table 2.



(a)



(b)

Figure 2: (a) Probability Density Function, (b) Cumulative Density Function

V. Graphical User Interface for Videoconferencing and Text Chatting

There may be a situation where patient is present in a rural hospital where sufficient expert doctors are not available. In such situations, doctor present in the remote hospital may need to contact an expert to take his advice for better treatment of the patient. Telemedicine is a branch of clinical medicine which deals with such treatments. In this paper, a graphical user interface for low cost videoconferencing for Online Health Care System is used, specifically to support consultations among remotely placed patients, rural health workers and specialists in the urban cities and provide a secure access to remote patient records.

Videoconferencing refers to the use of video and audio for the communication and interaction between individuals sitting at different locations. This paper also describes the use of a low-cost web-based videoconferencing system using free open source software components, a module for proposed Health Care System. We have used live video/audio, text chat and video recording/archiving tools and features for our videoconferencing.

Open Source and Other Software Components

We have made the videoconferencing platform using Red5, open source application and other applications such as Action Scripting, Adobe Flash CS3, J2EE and XML [4, 9, 11]. We have used Red5, Action Scripting and Adobe Flash CS3 for graphical user interface applications and J2EE and XML for recording.

Red5: Red5 is a free, open source software package written in the java programming language [13]. Its purpose is to support communication between Adobe flash applications. It has support for live stream publishing, audio and video streaming, object sharing as well as the recording of streams. Using Red5 we can make a working prototype videoconferencing application that works in the web browser. Red5 also support for user to broad a video stream live from a webcam for other users to view immediately or for the server to record for users to view later. Red5 supports RTMP (Real Time Messaging Protocol), a communication protocol developed by Adobe Systems for streaming audio, video and data over the internet between a flash player and a flash media server (FMS) [6,13].

Red5 server comes with a suite of server-side and client-side demo applications. This includes the OFLA (Online Open Source Flash Conference) application. We have utilized the source code two client side applications (Publisher and Recorder) to create our browse- side component for display our video streams. This component builds and displays a grid of webcam videos amongst users during a videoconference. This application can also record the video for archival. RED5 flash media server is open source and free. It offers same functionality as Adobe Flash Server. It is hosted on Google code and contained J2EE container and can host JSP and PHP pages. It also supports multiuser chat, live streaming and client stream recording. In Red5, we are using to develop browser based videoconferencing application. Here client uses Adobe Flash Player for capturing video and audio from players continuously media clips from web server.

Basic Parts of Videoconferencing System

There are total four modules used for the system.

1.Audio/Video Message Compression Module: This module is for controlling the quality of images and audio, and capturing the video and relaying it at the receiver's end. It deals with the transmission in the form of streams and frames.

2.File Sharing and Recording Module: This module facilitates the users to share files and record their video/audio conferencing and text chatting through internet.

3.User Management Module: This module is for authentication and identification of registered users. It also facilitates joining the user groups for audio/video conferencing.

4.Quality Control Module: This module is for controlling the quality of images and audio and capturing the video and relaying it at the receiver's end. It deals with the transmission in the form of streams and frames.

Work Flow

Figure 3 shows overall working flow of the videoconferencing system. The system has a webcam and a microphone attached to the computer through which input in the form of audio and video respectively is given. This input is given by the users through the user interface. The session of the user is maintained by the session manager. The data is transmitted through the transmitter via network. At the other end, receiver receives the audio and video which is handled by the audio/video handler. The output is shown through the monitor and the speaker. The graphical user interface is used to interact with application.

Figure 3: Data Flow in Videoconferencing System

VI. Working Snap shots of Videoconferencing using Red5 and Text Chatting:

Two snap shots of the videoconferencing and text chatting are as shown in Figure 4 and Figure 5 respectively.

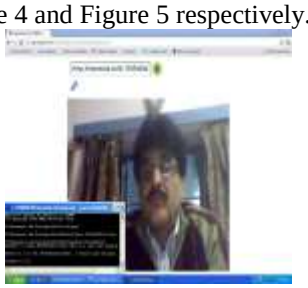


Figure 4: Snapshot of the Videoconferencing System at the Sender End

Figure 5: Snapshot of the Videoconferencing System at the Receiver End

VII. Conclusion and Future work

This paper has presented the development of a "Smart Card based Online Health Care System". This system helps in dealing with the frequent data storage, exchange and retrieval of data from the databases of hospital. It also helps the hospital system to retrieve the patient related information easily and efficiently from local or centralized database server using selection of appropriate memory management. It has also presented the use of graphical user interface for low cost videoconferencing for Online Health Care System. It is intended to support consultations among remote patients, rural health workers and experts in the urban areas.

Scope for future work:

1. A better with other needed facilities, a new Health Care System can be proposed.
2. Videoconferencing system with 3G and 4G mobile telephony and virtual reality technologies can be developed.
3. Using other databases such as MySQL and Ingres, application can be designed and performance can be analyzed.

References

[1]Chien-Tsai Liu, Pei-Tun Yanga, Yu-Ting Yeha, and Bin-Long wangb, "The impact of smart cards on hospital information systems: an investigation of first phase of national

insurance smart card project in Taiwan," International Journal of Medical Informatics,75(2), 173-181,2006.
[2]Michael Marschollek and Edip Demirbilek. Providing longitudinal health care information with the new german health carda pilot system to track patient pathways. Journal of Computer Methods and Programs in Biomedicine, 81(3):266-271, March 2006.
[3]D. Trcek, R. Novak, G. Kandus, and M. Suselj. The impacts of smart cards on hospital information systems an investigation of the first phase of the national health insurance smart card project in Taiwan. International Journal of Medical Informatics, 61(1):33-43, 2001.
[4] ASP.NET.http://www.asp.net.
[5]Vishal Charegaonkar, Kiran Hariharan Nair, and Gaurav Gautam. Smart card transaction processing-case study: Applying best practices to improve performance,IBM,Feberary2008.www.ibm.com/websphere/developer/zones/hipods.
[6]Lu Liu and XiaoGuo Dong. Red5 ash server analysis and video call service implementation. In Proceedings of IEEE symposium on web society, pages 397-400, 2010.
[7]Zou Libing, Wang Huajun, and Zhang Sanmei. An implementation of jabber in video communication based on Red 5. Journal of Information and Communications, 3, 2008.
[8] Tomcat Apache: The apache software foundation, 2009-11. http://tomcat.apache.org.
[9]Herbert Schildt. Java 2: The Complete Reference, Fifth Edition.2002.AdobeFlashPlayer.http://www.adobe.com/products/flashplayer.
[10]FlashActionscriptingTutorials.http://www.tutorialized.com/tutorials/Flash/Actionscripting.
[11]XMLtutorials.http://www.w3scools.com/xml/default.asp.
[12] Telemedicine in India: www.telemedindia.org.
[13]Red5:An open source ash server, 2005. http://www.red5.org.
[14]SQL Tutorial: www.w3schools.com/sql/default.asp
[15]Narendra Kohli and Nishchal K. Verma. Performance analysis of health care system. International Journal of Engineering, Science and Technology, 3(1):191-205, May 2011.

Biographical notes

Dr. Narendra Kohli was born in India on April 13, 1963. He received M. Sc. Engg. from Kiev university Kiev in 1988 and Ph.D. from Indian Institute of Technology Kanpur, India. He is currently an Associate Professor and Head with the Department of Computer Science & Engineering, Harcourt Butler Technological Institute Kanpur, India. He is a Member of IE (India), Fellow of IETE (India), and senior member of ISE.

Dr. Nishchal K. Verma was born in India on September 9, 1973. He received the B.E. degree from the Faculty of Engineering, Dayalbagh Educational Institute, Agra, India, in 1996, the M.Tech. degree from the Indian Institute of Technology (IIT) Roorkee, Roorkee, India, in 2003, and the Ph.D. degree from IIT Delhi, New Delhi, India, in 2007, all in electrical engineering. He is currently an Assistant Professor with the Department of Electrical Engineering, IIT Kanpur, India. His research interests include fuzzy systems, neural networks, data mining, fault diagnosis, bioinformatics, color segmentation, video clip or image sequence modeling, machine learning, and computational intelligence. Dr. Verma is a reviewer for several reputed national and international journals and conferences, including the IEEE TRANSACTIONS ON FUZZY SYSTEMS, the IEEE TRANSACTIONS ON SYSTEMS, MAN, and CYBERNETICS: PARTS A, B, AND C, the IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE, and *Pattern Recognition*.

Table 1: Retrieval Time with Traditional SQL Server Database

	On Asp.net Search Page (Front-End)		On Query Analyser	
Iteration	Execution time in ms using Stored Procedure	Execution time in ms using Query	Execution time in ms using Stored Procedure	Execution time in ms using query
1	.0491	.0543	.0564	.0645
2	.0473	.0537	.0554	.0589
3	.0489	.0631	.0572	.0675
4	.0491	.0571	.0602	.0614
5	.0478	.0544	.0546	.0598
6	.0483	.0597	.0617	.0678
7	.0462	.0549	.0545	.0559
8	.0497	.0572	.0621	.0656
Mean value In seconds	.0483	.0568	.0577	.0626

Table 2: Retrieval Time with Optimized SQL Server Database

	On Asp.net Search Page (Front-End)		On Query Analyser	
Iteration	Execution time in ms using Stored Procedure	Execution time in ms using Query	Execution time in ms using Stored Procedure	Execution time in ms using query
1	.0272	.0311	.0327	.0436
2	.0250	.0277	.0336	.0338
3	.0278	.0317	.0282	.0340
4	.0260	.0345	.0461	.0360
5	.0325	.0332	.0335	.0323
6	.0256	.0321	.0328	.0358
7	.0308	.0300	.0461	.0459
8	.0365	.0326	.0357	.0341
Mean value in seconds	.0289	.0316	.0361	.0369