자동 에이전트 기법을 활용한 임상문서저장시스템

論 文
54D-12-7

A Clinical Data Repository as an Autonomous Agent

金 和 仙† · 캠 비 즈* · 趙 勳**
(Hwa Sun Kim · Kambiz Houshiaryan · Hune Cho)

Abstract - In general, hospital information system should provide interoperability hat usually and operate independence of other HIS. This study proposes a new HIS paradigm that can be implemented within standard HL7 Interface engine and clinical data repository (CDR). We have developed an alternative architecture relying on agent solutions with distributed queries to heterogeneous databases. This architecture creates a very fine and flexible repository that can handle queries with the bases of standard HL7 messaging structure. Deploying Agent solutions to manipulate autonomy of storage management and sociality for communication with open world is another issue that keeps this system from reinventing existing wheels in medical informatics. This study the first attempt to construct CDR based private clinic. We used the information stored in the clinical patient record system of the internal medicine private hospital which is used rational database. We were searched increasing the 1,000 data entry from 1,000 to 10,000. By the result, experimental CDR showed highly efficient performance more than 6,000. In the future, the CDR can be further extended for clinical information among private hospitals estranged from EHR (Electronic Health Records).

Key Words : Clinical Data Repository, Health Level 7, Agent System

1. INTRODUCTION

A flexible clinical data repository is the one that can organize and store highly diverse and heterogeneous data with efficient performance which have been challenging issue for significant modeling and implementation of clinical repository systems. The entity-attribute-value (EAV) approach is popular for modeling highly heterogeneous data. We have enhanced this framework in the way that can handle HL7 messaging structure [8] and permit to be used with agent technology. In particular, attribute-centered queries, where the query criterion is based on the value of a particular attribute, are most likely to show impaired performance. This is especially true when query criteria combine one or more simple conditions in Boolean fashion and severely impaired performance when data transmissions are supported. This research aims to address some of these problems along with communication, sociality and autonomy of applications to upgrade existing systems in medical information technology.

† 교신저자, 正 會 員 : 慶北大學校 醫療情報學科 博士課程
E-mail : polala@knu.ac.kror
* 非 會 員 : 慶北大學校 컴퓨터科學科 博士課程
** 非 會 員 : 慶北大學校 醫科大學 醫療情報學科 主任敎授
接受日字 : 2005年 8月 13日
最終完了 : 2005年 11月 9日

2. MATERIALS AND METHODS

2.1 Architecture

The Basic architecture of current database system illustrated in figure 1. User defined tables available in HL7 V2.4 specification were referred to as a frame for architecture of CDR. A patient master index relates each stored patient medical history with different segments that owns a separate table with related attributes. Querying information will be asked to the system using related segment having empty fields as wanted information. As mentioned in HL7 V2.4 specification. if a field contains null value then the original value will be erased. To solve problems such as performance and query response time, firstly we included a file system database because accessing file system using MUMPS were faster than accessing centralized database [9]. In order to keep a patch related with existing history of medical record we grouped segments of a message and related them to a 3D master patient index-x with path of current record, y with fields in a segments and z with attached contents such as image or signal messages, which could improve the searching queries. Secondly, with cached storage we filtered outgoing array of results

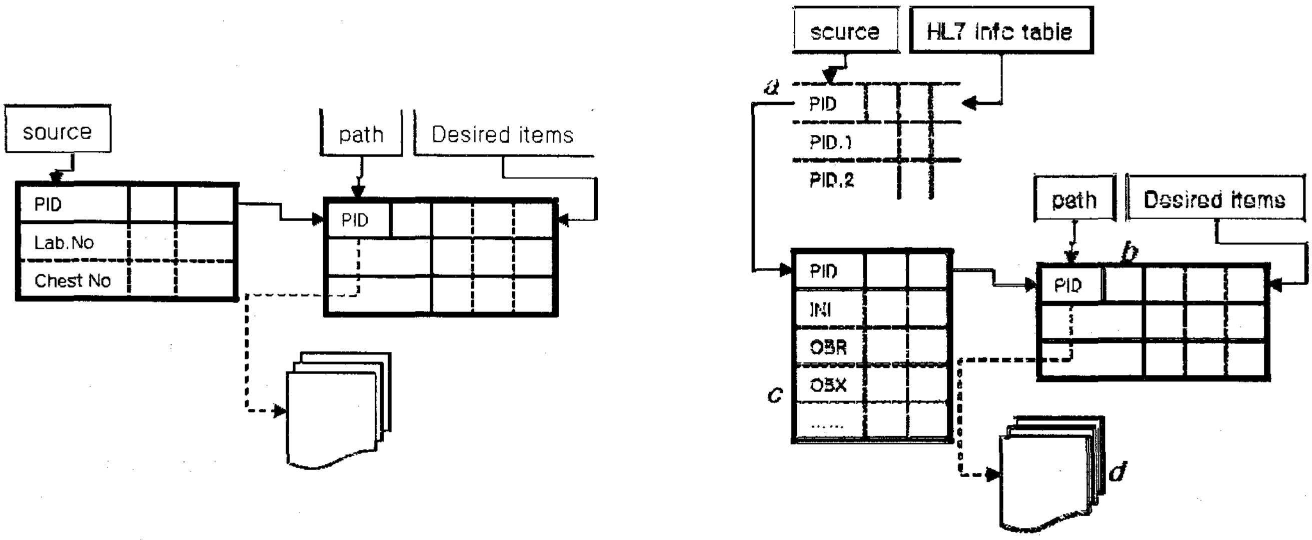

Fig. 1 A basic architecture of proposed hospital database system. a. HL7 standard and user defined table of segment information, b. Required patient information. Items related to a segment that must be viewed in first glance, c. Patient Master index. Information about current health record, d. History of patient medical records.

with Boolean queries for optimization, and reduced number of arrays to improve the performance so that the user could feel no delay time in searching system. Thirdly we separated the database into two parts; one for synchronized use defined as local database by putting away the transmission protocol, second for synchronize use defined as remote database that includes a transmission protocol and slightly delays the performance and as long as it is Asynchronous, delay time was not critical. Standardizing outgoing queries using HL7 messaging structure and wrapping them with envelops of FIPA (the Foundation for Intelligent Physical Agents) [10] standards for the ease of communication with different Agents were proposed to enhance the flexibility of communication. A component that could handle queries structured with HL7 message and could spread upon multi-agent platforms were added to response accordingly with given query message format. Figure 2 illustrates the relation between patient master index and wrapper, storage, medical vocabulary and segment indexing storage. Query for database will be queued from network as either agent language or unwrapped pure HL7 messaging structure. Each message will be unwrapped and split into segments. Each segment will contain fields that divides into two parts. One series of fields to be stored in segment indexing storage for the basic information of each segment and the rest of fields to be stored in database file system. Each message will have an index constructed with patient id and time of entry. This index will relate all segments of a message to a stored database which is patient medical history. Upcoming segments as result of each observation /procedure will be appended into the history of patient records.

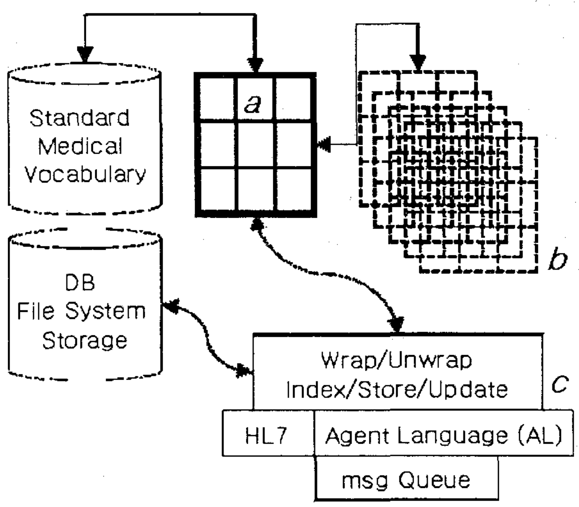

Fig. 2 System Architecture a. Patient Master Index, b. Indexing segments with fields and components, c. Language wrapper and indexing manager interface.

2.2 Implementation

Implementation of master patient index and standard of medical terminologies are constructed upon Microsoft access 2000. Basic storage of patient information is managed by file system database as EAV-described below and completed using part of MUMPS language. In an EAV database, each row represents one attribute-value pair. When a new attribute needs to be created in a conventional database, either a new column needs to be added to a table, or a new table needs to be created. In an EAV model, no schema changes need to be made, rather a new attribute name is used. EAV models have many advantages over relational databases, but have many problems including complex query performance. There are two main camps when it comes to EAV cores. One approach stores all values as text strings. This requires data types other than strings to be converted back-and-forth to text which is a problem when it comes to complex queries and leaves the data-type checking up to the application.

2.3. Agent and Autonomy

CDR consist of several software that are performing tasks through communications with different clinical departments. Performing a task, is an act that software does either with users command or through an unarranged decision. In order to modify our system to be an agent we have implemented and added a part that can reason about the requested queries. One of the query events for diagnosing a patient is from clinician to the storage of medical terminologies. Reporting results from existing storage will be a list of medical terminologies related to physician's input. A reasoning architecture has been added to the implementation of this part to optimize the result. Reasoning about input is to consider different situation that relates each input with existing terminologies and results in optimization of report. We have tested this and have gotten the result of 82% of correct results for each input. The rest of 12% will be viewed by physician's selection. In order to refer our system as an Autonomous Agent we designed a storage management to consider each entry either as an update for existing data or as an indexing input for existing storage. When the agent could not find any index related to new entry then it will create a new storage. Patching storage and considering a life time for upcoming data is the next procedure that our agent takes into the action. If data entry is indexed into master patient then it has a lifetime that during each, incoming data considered as an update for existing entry. A lifetime will be ended when the data is removed from the master index. In the other hand we considered storage management as an autonomous action of our clinical repository. Data entry will be managed and kept into local system for a specific duration, say year or two, depend on the hardware capacity. Overflowed data will be backed up and zipped. Alerting administrator for either enhancing the capacity or moving overflowed data to another system is the next task that our agent performs autonomously.

3. RESULTS

CDR as an autonomous agent can store, update and retrieve information to and from the database. Incoming queries from different agents, web based applications and client based application is acceptable to the repository system in the way that outgoing information is adjusted with incoming message format and can be used accordingly. A pure message can be documented according to different need as well. Table 1 and Figure 3 illustrate the performance test of systems. Line chart presented a smoother appearance than drawing bars over each number of entry point. We used the information stored in the clinical patient record system of the internal medicine private hospital which is used rational database.

Table 1 Legacy and new system performance results

System	Number of Entry					
	1,000			10,000		
	Search	Create	Update	Search	Create	Update
Legacy	1.00	1.02	1.45	5.00	4.00	3.20
New	1.02	1.03	1.44	4.00	3.11	2.40

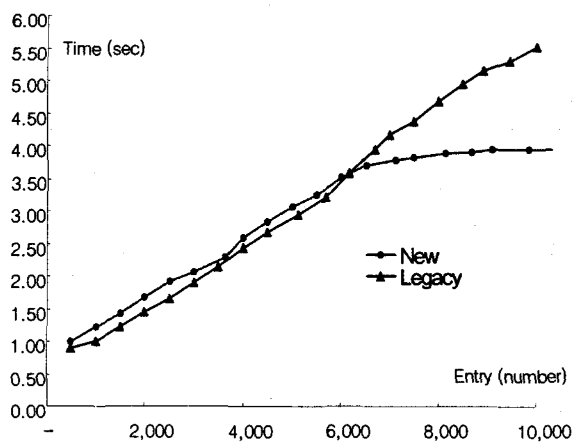

Fig. 3 Comparison of processing speed for increased Entry

We were searched increasing the 1,000 data entry from 1,000 to 10,000 and we have tested (search, create and update) each 3 times per 1,000 data entry and calculate average. By the result, experimental CDR showed highly efficient performance more than 6,000. For not more than 6,000, existing database was better/nearly performance.

In total proposed clinical repository system as an autonomous agent could store, update and retrieve information to and from the database. Incoming queries from different agents, web based applications and client based application were acceptable to the repository system in the way that outgoing information were adjusted with incoming message format which could be used accordingly. A pure message could be documented according to different need as well.

4. DISCUSSION

Contemporary medicine requires that an increasing quantity of clinical information be stored and readily available for retrieval in CDR. In most CDR more than one field can be expected to contain the data to answer a given question. This study had some inherent limitations. We looked at only one database and a limited number of performance test. And comparing to Legacy CDR, the goal, managing medical information, is same, but this CDR is broader CDR system instead of local CDR. So, The most important concept of CDR is interoperability because many hospitals have heterogeneous platform, system and operating system. And for the future work, semantic agreement will be derived and communicated. This kind of support of service is meaningful for shifting from component paradigm to service paradigm, following distributed environment, and achieving interoperability. Today, the needs of EMR, EHR are increasing rapidly, clinical information sharing and collaboration is necessary. In addition to those, clinical information sharing and clinical documents exchanging is required and that would be the reason of studying CDR. This CDR based system that manage and collaborate clinical documents will help to construct life-long electronic medical record environment faster and to improve the quality of care.

5. FUTURE WORKS

The problem that medical information is still suffering is the time of creation of new system which needs migrating clinical information from old database to new database. Because of wide variety of data storage used, among clinical care this migration is time consuming, very difficult, have many side effects and almost impossible for transformation. Because of this problem it is not easy to convince medical staffs of new systems. Although our work is to upgrade existing systems we are keeping our effort to solve above mentioned problem in near future. Since Medical Information Technology has been rapidly advancing and the amount of clinical data has been increased, developing efficient ways of managing and exchanging clinical information has become an important issue. The need to share clinical information has also increased the need to develop the infrastructure for the EHR system. This work suggests an approach for developing national infrastructure for creating EHR to improve the quality of healthcare and eventually reduce medical error.

ACKNOWLEDGMENTS

This work is supported by health industry development division of Ministry of Health & Welfare Korea (02-PJ1-PG6-HI03-0004).

The authors are grateful for support from Prof. Il Kon Kim and Researcher of IHIS in obtaining the data necessary for this study.

REFERENCES

[1] Winston PH. Artificial Intelligence. 2nd ed. Reading. Addison-Wesley, Reading, MA, 1986.

[2] Dwight J, Erwin M (eds). Special Edition: Using CGI. Indianapolis, Ind: Que Corporation, 1996.

[3] HL7, Available at: http://HL7.org, Accessed Dec. 24, 2004.

[4] John M. Lewkowicz The Complete Mumps An Introduction and Reference Manual for the Mumps Programming Language.

[5] Huff SM, Rocha RA, Solbrig HR, Barnes MW, Schrank SP, Smith M. Linking a medical vocabulary to a clinical data model using Abstract Syntax Notation 1. 1998, 440-452.

[6] Y. Labrou and T. Finin Towards a standard for an Agent Communication Language. Proceeding of the first international conference on Multi-Agent Systems, 1995.

[7] Fipa, Available at:http://fipa.org, Accessed May, 03, 2005.

저 자 소 개

Hwa Sun Kim, MS is received RN and has the M.S. degree in Computer Science and Engineering from Inje University. PH.D. student at Department of Medical Informatics, Kyungpook National Univ. Daegu, Republic of Korea. Her major interests and past accomplishments are design and implementation of XML based systems. Currently she has been working on "Architectural design of new paradigm hospital information system based on web service." Interested readers may contact the author, via either polala@knu.ac.kror Department of Medical Informatics at Kyungpook National University School of Medicine, Choong-gu Dongin-doon 2-101, Taegu Korea, 700-721.

Kambiz Houshiaryan, MS is received degree in Computer Science from Kyungpook National Univ. PH.D. student at Department of Computer Science, Kyungpook National Univ. Daegu, Republic of Korea. His major interests and past accomplishments are ontology and hospital information agent based systems. Interested readers may contact the author, via either kambiz@nate.com or Department of Computer Science at Kyungpook National University, Choong-gu Dongin-doon 2-101, Taegu Korea, 700-721.

Hune Cho, PhD is an associate professor of the Department of Medical Informatics at Kyungpook National Univ. School of Medicine. He was born and educated in Korea and further research was continued in USA. On his return, he teaches varieties of medical informatics courses and initiates research on information systems of Kyungpook University Hospital. In addition, he is currently involved in international standardization of medical information for ISO/TC215. His major resarch interests are hospital information systems, medical imaging, computerzied patient record, web-based telemedicine and many other health related issues. Interested readers may contact the author, via either hunecho@knu.ac.kror Department of Medical Informatics at Kyungpook National University School of Medicine, Choong-gu Dongin-doon 2-101, Taegu Korea, 700-721.