Intelligent Storage Devices for Mobile Databases

N. Mallikharjuna Rao

***Abstract*—In our previous work, we proposed an intelligent architecture for mobile system, intelligent processing, querying and accessing of huge data for mobile databases. In this paper, we proposed intelligent storage devices for handling huge data on mobile databases. This is a massive storage resolves the capacity requirements of huge data storage in the networks of enterprise and organizations. The complexity and cost of storage management needs intelligence to reduce the huge overheads, and intelligent storage becomes the ideal storage solutions. In this paper, we illustrated architecture for intelligent storage, scheduling policies and key factor of intelligent devices.**

***Index Terms*—Intelligent storage devices, technologies, virtual storage, object based storage, network storage, fuzzy databases**

I. Introduction

The Telecommunications system is rapidly increasing day by day in India and adding Millions of subscribers in every year and even months. The Government of India is planning to integrate to all the service providers' database in future that means it required huge database space for future data. For example, the competition between the telecom service provider's, they are bringing SIM cards with very cheep cost for attracting mobile Subscribers, hence subscribers are increasing every day. For organizing such data we are using traditional database systems, may be in future it is difficult to handle with traditional relational databases. Data is created and coming inform of unstructured and even it is difficult to retrieve with relational database queries on mobile databases. In future we face a problem that is, how to manage mass unstructured storage systems and massive database of mobile service providers becomes a critical situation.

In PCS networks, we have to maintain location databases along with subscriber's database. General methods require a mobile to report its location to the network using different criteria [1]. The network stores the location of the mobile in location information databases and this information is retrieved during call delivery. When a LID fails, call may have to be dropped as the location information of mobiles registered in the LID is occupied. The failure of some of these databases, therefore, negatively affects the performance of the networks. All such type of transactions are required to save on databases

In GSM standards there are two types of database; one is a global Home Location Register (HLR) and a Local Visitor Location Register (VLR) to store the location information of mobiles. When subscribers are calling /retrieving the database from the mobile database is a single point of failure. As the number of mobiles increase in the system the load on databases is also a factor which decides the call setup and retrieval time.

Manuscript received June 22, 2012; revised July 15, 2012.

N. Mallikharjuna Rao is with the Annamacharya P.G College of Computer Studies, Rajampet (e-mail: drmallik2009@gmail.com).

The storage intelligence and how to construct intelligent system, there are not general criteria yet, and the key factors of intelligent storage should be paid more attention. Intelligent storage may also be called smart storage. All intelligent are from human beings, from the designed architecture of the people, and the collaboration of the compositions [2]. The paper focus on the architecture and key factors of intelligent storage or smart storage more about the algorithm and distributed environment on mobile devices.

This paper is organized as follows. In Section 2 we take a brief look at related work with fuzzy database and fuzzy storage devices, Section 3 illustrates the architecture of intelligent storage devices. In Section 4 discussed some key factors and scheduling algorithm of intelligent storage is explained. At last a conclusion is given.

II. Related Work

In our previous work, we proposed to implement call registration and call delivery algorithms for Home Location Register (HLR) and Visitor Location Register (VLR) with fuzzy database approaches for reducing the transition time between the HLR and VLR call registration and call delivery. In another work, we presented how the call registration and call deliveries are registered and stored in database values stored in database intelligently discussed in [3] and how the records are stored and present in the form of structures in the fuzzy databases. In this work, we also described the how the fuzzy crisp and variable behave for mobile databases. In another work [4] [5], we illustrated the intelligent data structures for mobile devices, introduced the record structure and file organization schemes for fuzzy databases.

Based on the above information, for organizing the huge data on mobile devices we need the smart or intelligent storage devices for keeping data long while. In VLSI and embedded Operating Systems, the architecture of storage system has gradually changed. The traditional storage systems such DAS (Direct Attached Storage), NAS (Network Attached Storage), of SAN (Storage Area Network) their architectures are predefined. OBS (Object-Based Storage) or OSD (Object Storage Device) pass on storage management down to storage devices with some intelligence, and the intelligent storage will dynamically adjust itself to optimum and load balance status. This study [6] can be help to make and propose the intelligent devices for storing the intelligently and decrease the access time and increase the search, insert and delete operation time.

DOI: 10.7763/IJET.2012.V4.494

A. *Direct Attached Storage*

Direct-attached storage (DAS) is a basic level storage and the first storage model that gained widespread acceptance and is widely used. Even though networked storage models are gaining popularity, DAS still finds takers because it is easy to deploy and has a low initial cost of deployment. It helps if you have an idea of what your data availability needs are at present and what they will be in the future. Since clients can access the storage device only through the server, a high percentage of server uptime is critical. A slow server will make storing and retrieving files difficult. Since the server also runs applications, data access may be slow as server bandwidth gets diverted to the applications.

B. *Network Attached Storage*

Network-attached storage (NAS) is file-level computer data storage connected to a computer network providing data access to heterogeneous clients. As of 2010 NAS devices are gaining popularity, as a convenient method of sharing files between multiple computers. Potential benefits of network-attached storage, compared to file servers, include faster data access, easier administration, and simple configuration. NAS systems are networked appliances which contain one or more hard drives, often arranged into logical, redundant storage containers or RAID arrays. Network-attached storage removes the responsibility of file serving from other servers on the network. They typically provide access to files using network file sharing protocols such as NFS, SMB/CIFS, or AFP.

C. *Storage Area Network*

A storage area network (SAN) is a high-speed special-purpose network that interconnects different kinds of data storage devices with associated data servers on behalf of a larger network of users. Typically, a storage area network is part of the overall network of computing resources for an enterprise. A storage area network is usually clustered in close proximity to other computing resources such as IBM z990 mainframes but may also extend to remote locations for backup and archival storage, using wide area network carrier technologies such as ATM or SONET.

D. *Object Based Storage*

An Object-based Storage Device (OSD) is a computer storage device, similar to disk storage but working at a higher level. Instead of providing a block-oriented interface that reads and writes fixed sized blocks of data, an OSD organizes data into flexible-sized data containers, called *objects*. Each object has both data and metadata (an extensible set of attributes describing the object). The command interface to the OSD includes commands to create and delete objects, write bytes and read bytes to and from individual objects, and to set and get attributes on objects. The OSD is responsible for managing the storage of objects and their metadata. The OSD implements a security mechanism that provides per-object and per-command access control.

E. *Storage Grid*

"The Network is the Computer", we can say the network is storage, such as web pages, e-mails, database, Mobile subscribers data etc., are all stored in the storage devices. No storage, no network. Storage grid includes intelligence such as migrating the data and the components in the network to share the data and the codes. Storage grid is the practical intelligent storage and it may gather the above referred storage systems and build a virtual huge storage pool to supply infinite storage ability.

III. Intelligent Storage Devices Structure

The development of the massive storage and the complexity of storage management, the cost are increased rapidly. The intelligent storage system is needed to reduce the total cost Own (TCO) , so we think that an intelligent storage is appropriately designed and should provide transparent interface, active data allocation, flexible and efficient storage management, etc. the goals are high performance, high scalability, high availability, high reliability and high security for maintaining the databases on mobile databases.

In this paper, we proposed intelligent storage devices which are used for storing and saving the subscribers and other transactional data for mobile phones or devices. Increasing bandwidth of networks and several subscribers are spread the unstructured data from one mobile to another mobile, remote access to network various storage systems becomes reality. Because of all the problems which we face with unstructured data to handle on mobile devices, to minimize the collaboration and interactive cost, distributed integrated intelligent storage system based on multiple agents and object storage devices and virtual storage is proposed as we shown in Fig. 1, which is a hierarchical layered structure.

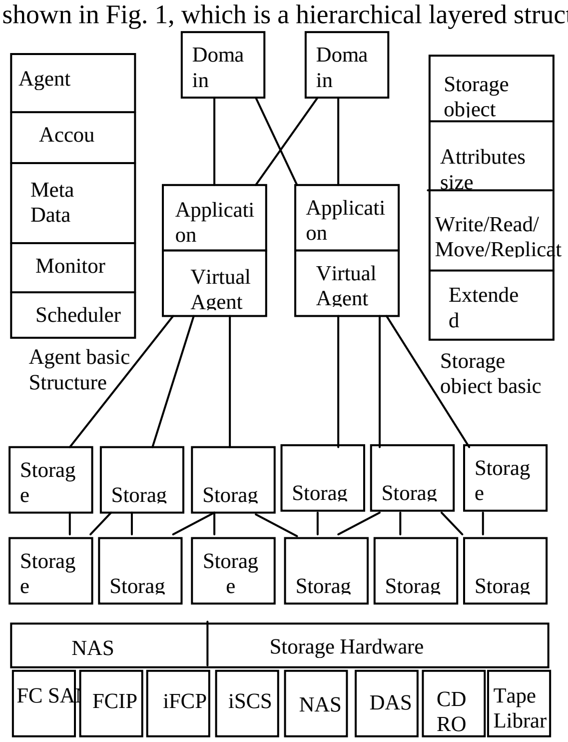

Fig. 1. Intelligent storage system architecture.

Self-similarity of agents can simplify management operations and reduce interconnection cost. The agent layer is divided predefined or constructured on demand. Intelligent storage takes storage as basic intelligent units with inner intelligence and clusters the storage objects to construct course intelligence and add more overlay agents and extracted common behaviors to collaborate in access and control.

Storage administration is the core of the storage intelligence. The intelligence of self- management represents ability of the intelligent storage. Generally, storage administration can be implemented by the operating system outside in the special server, or bye the embedded operating system inside the storage system. The original storage devices are often managed by the server, which is slowly, while the intelligent storage device is usually managed by its own controlling codes.

Intelligent storage devices have a great advantage which is an abstraction between servers and storage made it possible for the server to use storage without any direct knowledge of exactly what it was or how it was configured. This was a great thing for the industry because it allowed storage interconnect standards such as Fiber Channel and iSCSI to allow interoperability between just about any complaint server operating system and storage array. Storage vendors have come out with proprietary, server-based software to allow servers to work more intelligently with storage. These devices are allowed direct traffic to the most advantageous storage controller port, and transparently work around fabric failures.

IV. Key Factors

Storage administration is the core of the storage intelligence. The intelligence of self-Administration represents ability of the intelligent storage. The following are the key factors for storage administration on devices for mobile databases.

A. File System

File system is the highest administration stage than storage block, and it can manage the storage resources more efficiently. File system is so important in intelligent storage devices which be can mapped to the physical storage device for storing file list of tables in file system directory. In this file system of intelligent storage may contain more attributes to help with administration storage layout and security and also reduce latency and seek time of data records from the devices.

B. Back Up and Restore

Very important thing in this proposed intelligent storage should support back up and restoring data realistically and without human interference. Operation should not affect the normal applications. Back up and restore cab be remote and guarantee the responding time. It is carried by metadata servers or specified data backup and restoring components.

C. Security Intelligence

Intelligent storage system is often constructed from many storage devices distributed in network, such as internet, mobile networks. In storage devices, the security of intelligent storage exits in every aspect of data communication and share phases. iSCSI is increase the interaction between storage devices and standard protocols like SCSI. This will increase the consistency on storage devices for accessing and storing the data.

D. Storage Sharing

An I/O architecture permitting sharing enables seamless fail-over of the storage-consumers that share data sets. Thus storage of data sharing is essential to provide continuous service. Shared storage architecture also provides better scalability since storage capacity and processing power could be added dynamically to the pool by adding more storage-consumers and storage. Data sharing given high flexibility for dynamic load balancing since the data is uniformly accessible from any storage consumer. Sharing also facilitates storage consolidation which reduces management and operation costs while increasing system usage. In traditional device there is a I/O fencing problem. In neither FC nor iSCSI provide any concurrency control primitives they do not provide I/O fencing.

E. Throughput

With high speed network and disk interfaces, SANs are expected to be able to move the data as fast as possible. As the carrier bandwidth increases, the transport protocol inefficiencies at the end-points have a negative impact on the effective delay and throughput. FC SANs have better throughput than iSCSI in local environments; any how it is fast, efficient, and simple and has better flow control support.

F. Availability

Several intelligent storage devices are available for organizing the imprecise data. FC SAN, iSCSI, iFCP and FCIP these are the devices which are used for organizing the data on networks or even on local places. For example, as the SAN infrastructure it is having primitives that help storage consumer to ensure data availability. Multicast is one such primitive that helps in proving availability using replication. The properties provided by the traditional hardware multicast are not sufficient to ensure the mutual consistency of replicated copies. If a SAN is to provide stronger multicast guaranties, it will ease the effort needed in proving transparent, block level replication.

G. QoS Administration

QoS has been used in the Web based Services and network applications and especially used in mobile networks and mobile applications for improving the quality of service on accessing, searching records on mobile databases.

V. Scheduling for Intelligent Storage

In this paper, we proposed intelligent storage devices for mobile database for handling the location information databases and other databases on mobiles like HLR and VLR for maintaining the subscribers profile while they are connecting to other subscribers or transferring the data at different places. In our previous work [4] [5], we illustrated fuzzy based notation of data on mobile devices. In this study, we used for storing intelligent storage devices to handle such database. In Section 3 we discussed architecture of intelligent devices and key factors discussed in Section 4. Another important aspect of storage device is disk scheduling process for improving the seek and response times of records.

In this paper, we are focusing in fuzzy-based approach for disk scheduling optimization. We applied fuzzy logic to disk scheduling policy, hence this algorithm is embedded firmware added to the hard disk controller. This approach takes for beginning of the sector to reach the head is known as rotational delay or rotational latency. The sum of the seek time ans the rotational delay equals to access time. In general, we studied so many scheduling policies [6] [7] [8] [9] like, First Come First Serve (FCFS), Shortest Seek Time First (SSTF), SCAN scheduling policy, Cyclic-SCAN (C-SCAN), LOOK and Cyclic –LOOK (C-LOOK). These are the various types of scheduling policies which can be used to access the records from the memory location in operating systems also provide and issue for providing a quick response time in query processing on devices.

A. Implementation of Scheduling Algorithm

In this study, we took First Come First Serve (FCFS) policy for implementing fuzzy based scheduling disk algorithm [10] [11]. It is using MATLAB with Fuzzy Logic toolbox.

The process starts using the processor it will continue till it finishes. The fuzzy inference system will interpret the values in the input vector and based on the set of rules assigns values to the output vector. Our fuzzy logic based system will consist of two inputs that is seek distance , *d* and waiting time , *t* and one priority output as shown in Fig. 2.

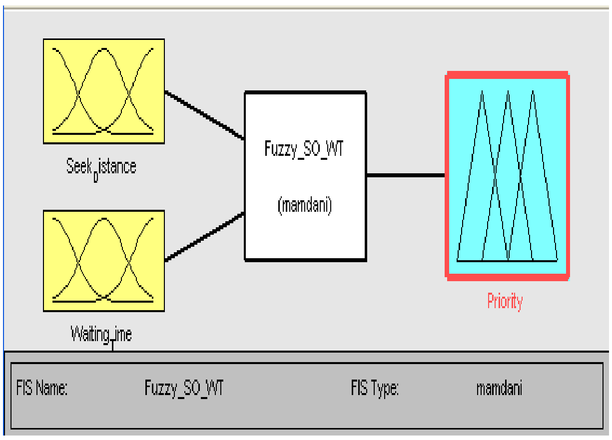

Fig. 2. Fuzzy system for disk scheduling

As we shown in Fig. 2, the two inputs are defined as:

1) Seek distance, *d:* the distance from the current head arm position to the next position.
2) Waiting time, *t*: the time the requests waits in the queue from the time it enters it till the time it starts getting service.

As the seek distance and waiting time of the certain requests (inputs), and it will produce output which is optimized priority with the help of set of rules which can be defined by membership functions. The output will be optimized priority. For the output, optimized priority, 6 memberships function with gaussmf type has been specified. The range is from 1 to 10 and 10 is the highest priority. A set of rules has been created as we shown in Fig. 3 which is give priority of scheduling for disks.

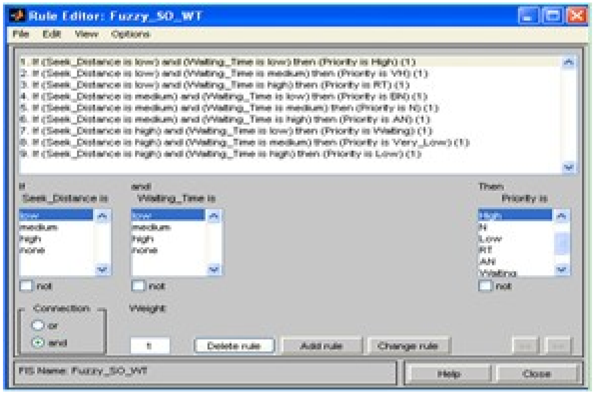

Fig. 3. Fuzzy rules for disk scheduling policy.

In summary, a number of requests that are in the queue will have an estimated seek distance which will be given as an input to the fuzzy logic model. We will assume that the requests in come one after another. The seek distance and the waiting time will be input to the fuzzy logic model, which in turn will give the output of priority requests. In Fig. 4 we have shown the performance for priority as per the fuzzy set rules. Rules which made with 6 membership functions are as follows:

{VeryLow, Low, LowMedium, HighMedium, High VeryHigh}

Fuzzy based algorithm is having the advantages of overcome starvation and fairly treats all requests from the same point of view.

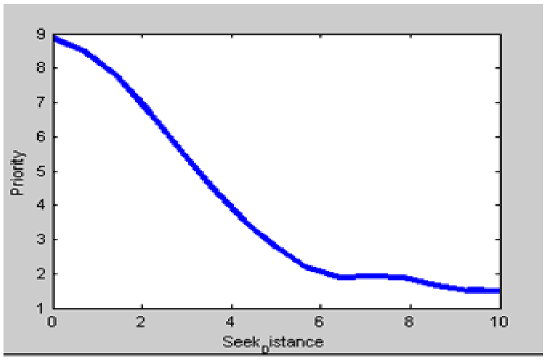

Fig. 4. Fuzzy based chart for seek distance.

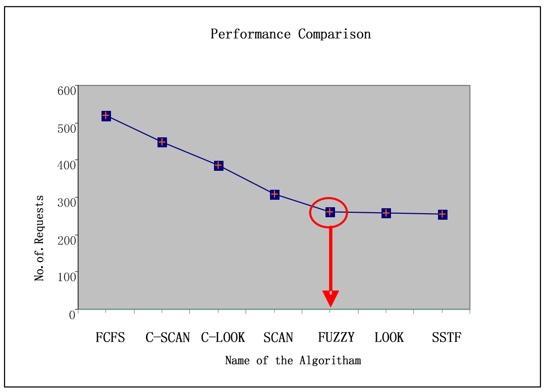

Fig. 5. Comparisons with other algorithms.

B. Comparison Result

In this work, we have presented the comparison of performance of this fuzzy logic model with the traditional algorithms. We took simple example, assume a disk wit h200 tracks and that the disk request queue has random requests in it. The requested tracks, in the order received by the disk scheduler, are 55, 59, 38,19,89,159,29 and 185 and also assume the initial waiting time is to be one. Using this two attributes we can apply to the fuzzy systems to come out with priority. After finished, line chart can be plotted to compare the performance of the fuzzy based FCFS and traditional algorithms as we shown in Fig. 5.

Seek distance movement is the performance metric for this chart. As we can see, the fuzzy algorithm can easily beat FCFS algorithm by extreme difference. However, in this results, LOOK and SSTF still came with better results even the difference is very close for these two algorithms.

VI. Conclusion

Intelligent storage devices are an emerging storage system for huge mobile databases and enterprise data management. In this paper, we presented intelligent storage device architecture and intelligent storage devices to maintain huge database of all mobile subscribers. Storage management and service may be critical requirements about storage performance and efficiency. For increasing performance in intelligent storage devices we illustrated fuzzy based disk scheduling for decreasing the seek time and waiting time. It will help for Cloud storage which provides pervasive and easy access based on internet in a new trend.

References

[1] G. Krishanmurthi and A. K. Somani, “Effect of Failure on Optimal Location Management Algorithms,” Lowa State University, Ames IA 50011.

[2] P. Bosc and H. Prade, “An Introduction to fuzzy set and possibility theory-based approaches to the treatment of uncertainty and imprecision management system,” In second UMIS workshop (Uncertainty management in information systems: from needs to solutions), Catalia, 1994.

[3] P. Bosc and M. Galibourg, “Indexing priciples for a fuzzy database,” *Information Systems*, vol. 14, no. 6, pp. 493-499, 1989.

[4] N. M. Rao, M. M. Naidu, and P. Seetharam, “An intelligent data structure approaches for mobile devices,” *IEEE, the 13th International Conference on Advanced Communication Technology*, February 13-16, Phoenix park, Korea (South).

[5] N. M. Rao and P. Seetharam, “An intelligent software workflow process design for location management on mobile devices,” *International Journal of advanced Computer Sciences and Applications*, ISSN: 2156-5570, vol. 1, no 5, November 2010.

[6] G. Narayan and K. Gopinath, “ISAN-An intelligent storage Area Network Architecuture,” L. Bouge and V. K Prasanna: HiPC 2004, LNCS 3296, pp. 262-273, 2004.

[7] D. Feng and H. Jin, “Massive storage systems,” *Journal of Computer Sciences and Technology*, vol. 21, no. 5, September, 2006, Special issue dedicated to the 20th Anniversary of NFSC, pp. 648-664.

[8] W. Stallings, “Operating Systems: Internal and Design Principles, 6 Edition,” Prentice Hall, 2009.

[9] Hofri and Micha, “Disk scheduling FCFS Vs SSTF revisited,” *Communications of the ACM*, Nov. 1980, vol. 23, no. 11.

[10] W. C. Oney, “Queueing analysis of the scan policy for moving head disk,” *ACM* vol. 22, no. 3, pp. 397-412, 1975.

[11] M. S. A. Talip and A. A. Aburas, “Fuzzy Logic Based Algoritham for disk scheduling Policy,” *2009 International Conference of Soft Computing and Pattern Reconnection, IEEE* , 2009.

N. Mallikharjuna Rao is presently working as Associate Professor in the Department of Master of Computer Applications at Annamacharya PG College of Computer Studies, Rajampet and having more than 12 years of Experience in Teaching UG and PG courses. He received his B.Sc Computer Science from Andhra University in 1995, Master of Computer Applications (MCA) from Acharya Nagarjuna University in 1998, Master of Philosophy in Computer science from Madurai Kamarj University, Tamilnadu, in India and Master of Technology in Computer Science and Engineering from Allahabad Agricultural University, India. He is a life Member in ISTE and Member in IEEE, IACSIT. He is a research scholar in Acharya Nagarjuna University Under the esteemed guidance of Dr.M.M.Naidu, Dean, Dept of Computer Science & Engineering, SV.Univeristy, Tirupathi, and AP.