Data Split and its Effect on Optimizing the Database Access Time in Mobile Networks

Mehdi Samie, Hassan Moradi, and Nafiseh Saberi
Iran Telecommunication Research Center, Tehran, Iran

Abstract: *In this paper, we present a method that reduces access time to subscriber database in mobile networks. At first, we assume a model to study the mathematical analysis about database and then we will show that this method can optimize the access time to database. So, we analyze related concepts and present an analytical model to formulate service time for a database. Also, we have studied and analyzed the effect of environment on the access time to database which is reviewed here. The environment is mobile network and the database is a mobile system database. Mobile network has some of special parameters which define both the rate and type of queries to database. We have analyzed environment parameters and formulated the access time to database based on them. Our method is data split. Compared to flatly storing of data, this method is influenced by the new environment parameters. These parameters include rate of access to split parts, split ratio of data and etc. We formulated our method and showed reduction of access time. Furthermore, we present some diagrams which are showing the access time versus split ratio. These diagrams also are showing conditions and limitations which split ratio optimizes access time. Finally, we have simulated our idea and observed that simulation results confirm theoretical results.*

Keywords: *Mobile database system, mobile network, subscriber database, data split, access rate, access time, service time.*

Received July 28, 2005; accepted June 14, 2006

1. Introduction

In mobile network, user's information is stored in different levels of database. In the current network based on European standard GSM, subscriber database has two levels: Visitor Location Register (VLR) and Home Location Register (HLR) and information stored on them contain common services, user location and etc [3]. Access time to database has an important role in network efficiency, because database access is necessary for each location updating or call delivery. Accesses to all stored information in database aren't in the same priorities and rates. Some information is used frequently and some is used rarely [2].

In subscribers' point of view, evaluation factor of database is the average of response time to requests so that shorter time shows higher efficiency. In this paper, regarding the operation of database in network, we have chosen a simple model so that based on the model and database functionality in details, we have generated mathematical formulas for access time. We have also proposed a method for storing information in the database. In this method, the stored data is split into two parts. Because of the access rate and based on the results of mathematical analysis, it defines conditions which optimize access time to database. We have simulated the database model, based on [3] for visitor location register and presented simulation results. These results confirm access time optimization.

2. Datbase Model in Network

At the first, it is necessary to evaluate the network from the viewpoint of database and determine which parameters influence access time for formulating database access time. So, a simple database model is the first step for reaching analytical model. As shown in Figure 1, the model contains request generator, server, database, and service queue. Request generator generates various requests and sends them to the server. Requests are queued up in the service queue. When server receives a request, it makes a connection to database through I/O device.

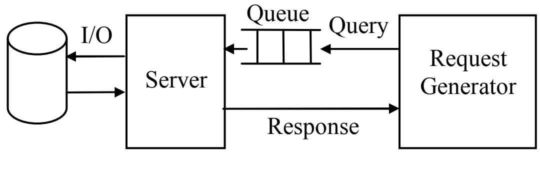

Figure 1. The simulated model of database.

3. Analytical Model

Efficiency of proposed method is proved in an analytical model before being shown by simulation. The analytical model is based on the database model in previous section. In this model, various parameters influence the database access time in network, which are ignored in analytical model. The main targets of mathematical analysis are evaluation of proposal method and comparing it with current method in theory, so that ignoring these parameters doesn't affect the comparing results.

In the analytical model, database access time depends on environment properties, request properties and database functionality. Environment properties include causes of requests generation and the rate of their arrival. Request properties include requested or updatable titles in queries. Database functionality also includes queue and its specifications, structure of indexing and buffering technique to access data on storage media.

3.1. Queue Model in Database

Every database can be modeled as a server with a buffering queue. Requests are processed based on FCFS in database server. The queries served by database are involved in two parts: Data request and data update [6]. For simplification, we assume that service time for both parts is the same, the waiting queue for queries is unlimited and number of requests for database services is a lot due to our requirement in future. We can readily see in [9] that with these suggestions, traffic of requests can be approximated as Poisson distribution. If service time of database is also approximated as a general distribution, database can be modeled as an M/G/1 queue, based on queue theory in [5]. If the arrival rate to queue be λ, the average of the access time denoted by W is as follows:

$$W = \bar{x} + \frac{\lambda \overline{x^2}}{2(1-\lambda\bar{x})} \tag{1}$$

Where $\bar{x}$ is the average service time of the database and $\overline{x^2}$ is the second amount of it [9]. General distribution for service time can be assumed as a uniform distribution in every database, so

$$\overline{x^2} = \frac{4}{3}\bar{x}^2 \tag{2}$$

3.2. Queue Model in GSM

In GSM standard, the rate of requests to database can be calculated as follows [9]:

$$\lambda = \gamma_r + 2\gamma_u \tag{3}$$

Where γ_u is the rate of location updating and γ_r is the rate of call delivery, so with using (1)

$$W = \bar{x} + \frac{\frac{4}{3}(\gamma_r + 2\gamma_u)\bar{x}^2}{2[1-(\gamma_r + 2\gamma_u)\bar{x}]} \tag{4}$$

3.3. Buffering in Database

Database systems use buffering technique to reduce the service time of database. This method reduces disk I/O. In this technique, database information is divided into disk pages with the same size. Due to memory limitations, only a few part of database pages can be resident at once in memory buffers. All database systems retain disk pages in memory buffers for a period of time after they are accessed by application [8]. An important problem arises, when a page is read from disk and all current buffers are in use. When a new buffer is required, one of the buffer pages should be dropped from memory and replaced by the new disk page. Which page should be dropped? It depends on page replacement algorithm which is used in database system. Some of page replacement algorithms can be seen in [1]. The best algorithm for page replacement is Least Recently Used (LRU). Based on LRU algorithm, the page which is the least recently used drops from memory.

In any way, memory buffers are reviewed firstly. If the requested page be in buffers, they are accessed, otherwise, they are read from disk and retain in buffers for a period of time. So, the time to take the information which depends on the time of catching both disk and memory, can be calculated as follows:

$$\begin{aligned} \bar{x} &= P_{miss}(T_{pd} + T_{pm}) + (1-P_{miss})T_{pm} \\ &= P_{miss}T_{pd} + T_{pm} \end{aligned} \tag{5}$$

Where P_{miss} is the probability of a situation when the requested data isn't in memory buffer and there is a need for disk access, T_{pm} is reading time of buffer and T_{pd} is the reading time of disk page.

Let *n* be the number of database disk pages and *m* be the number of memory buffers. In LRU algorithm, the required page will not exist in memory if this page isn't referred in *m* recent reference to database. If β be the page reference probability between *n* pages, P_{miss} is as follows:

$$P_{miss} = (1-\beta)^m \beta \tag{6}$$

We can assume that access to disk pages of the database is in the same priority, so

$$\beta = \frac{1}{n} \tag{7}$$

3.4. Data Indexing

One of the most important effective subjects in database access time is the method of data indexing, the way which data is indexed. Clustered-B-tree is the newest method of indexing in database systems. Access time in this method is shorter than non-clustered or other methods [10]. In this method, based on the selected key for each table we have:

1. A B-tree is created.

2. The table's records are sorted physically. In this case, each cluster can be accessed by one of index entries. In clustered-B-tree method, a group of records which have been placed in one page of disk, are introduced by one index entry in B-tree. In fact, database disk pages are leaves of B-tree. Each B-tree leaf contains both data records and index records. In B-tree structure, none of the nodes has a free space more than half node and if it happens, this page will be merged with other pages. So, the free space in leaves of B-tree such as other nodes is approximated to the $1/4$ of page space [10].

In this paper, for simplification, we assume B-tree nodes, which contain only index records, are resident in memory because of frequent access. So, the time of parsing the tree for non-leaf levels is ignored. This time is not effective in our comparison for final result. Both subscriber records in the database and disk pages in the B-tree are in the same priority. We suppose the records are identically distributed in pages. Let *N* be number of subscriber records and *P* be size of disk page or memory buffer page and *R* be size of each subscriber record. Access probability of each page of disk denoted by β is as follows:

$$\beta = \frac{3}{4}\frac{P}{N.R} \qquad (8)$$

Finally, the database service time in analytical model can be derived as follows:

$$\bar{x} = \frac{3}{4}\left(1-\frac{3}{4}\frac{P}{N.R}\right)^{m}\frac{P}{N.R}T_{pd} + T_{pm} \qquad (9)$$

With the assumption of present database queue, *W* can be calculated from (4) and (9).

4. Data Split and Optimizing Access Time

4.1. Access Time and Data Quantity

Equation (9) shows the relation between the database service time and other parameters. Equation (6) is a geometric function that has a maximum value for $\beta = 1/(m+1)$. The worst case for database service time occures when the maximum value happens. In this case, β is calculated as follows:

$$\beta = \frac{3}{4}\frac{P}{N.R} = \frac{1}{m+1} \qquad (10)$$

So, we have the following:

$$N.R = \frac{3P}{4}(m+1) \qquad (11)$$

Where *p* and *m* specify memory conditions. Specially *m* depends on allocated memory and a large quantity of it reduces database service time. Figure 2 shows the service time of database versus quantity, *N.R* value where T_{pd} is equal to *100msec* and T_{pm} is *10nsec*. Below diagrams have been presented for various values of *p* and *m*, assigned to typical value.

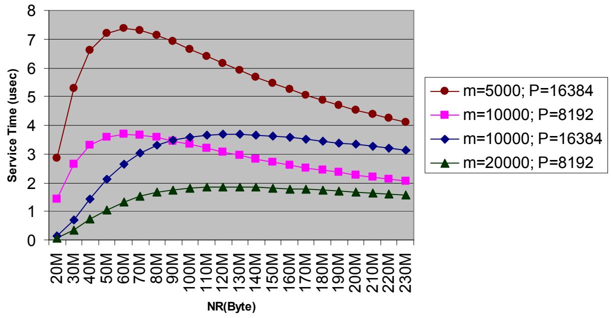

Figure 2. The database service time versus *N.R*.

By Figure 2, we can observe that larger memory buffer leads to lower service time and higher number of page buffers leads to either lower or higher service time, depending on the data quantity value. Also, we can observe that the service time of the database at first increases as the data quantity increases and decreases after a quantity value so that the diagrams have a maximum point. The most important reason for this specification is buffering algorithm. With comparison between diagrams we can observe the effect of changes for *p* and *m* or both on database service time.

4.2. Data Split and Access Time

In this paper, the target is the review of data split's effect on the access time and also review of the conditions which optimize the response time. Data split is based on the rate of access and information splits in two groups: *Hot* and *cold*. The idea for this naming has been taken from [4]. In comparison with cold group, hot is accessed in higher rate. In usual conditions, when there is an access to subscriber data, all of related information to subscribers is loaded into memory. Our purpose is to avoid loading cold data into memory while accessing to hot data which are accessed in higher rate.

Anyhow, data split has no effect on the number of records and it is only effective on leaves that contain records in clustered-index structure. Data split reduces the size of information that is resident in memory buffers because the part of subscriber's information that is always used, is often resident in memory. This manner combining with buffering algorithm decrease disk I/O. Amount of disk I/O reduction and consequently the access time depend on data split ratio (or in the other words subscriber's record split ratio) that is the ratio of hot or cold parts to whole data. Equations show that data split with each ratio doesn't optimize access time and based on network conditions, this ratio is adjustable. In case of data flat that we don't have any data split, based on (4) and (9), database access time denoted by W_{flat} is as follows:

$$W_{flat} = \alpha_{flat} T_{pd} + T_{pm} + \frac{\frac{4}{3}(\gamma_r + 2\gamma_u)(\alpha_{flat} T_{pd} + T_{pm})^2}{2[1 - (\gamma_r + 2\gamma_u)(\alpha_{flat} T_{pd} + T_{pm})]} \quad (12)$$

Where

$$\alpha_{flat} = \frac{3}{4}(1 - \frac{3}{4}\frac{P}{N.R})^m \frac{P}{N.R} \quad (13)$$

In case of data split, access rate to hot and cold groups is effective in access time. Let P_h be the access rate to hot group and P_c be the access rate to cold group. The average of the database service time can be derived as follows:

$$\overline{x} = P_h \overline{x}_h + P_c \overline{x}_c \quad (14)$$

Where $\overline{x}_h$ and $\overline{x}_c$ are relevant to hot and cold separated database tables and (5) satisfies them separately as follows:

$$\overline{x}_h = P_{hm} T_{pd} + T_{pm} \quad (15)$$

$$\overline{x}_c = P_{cm} T_{pd} + T_{pm} \quad (16)$$

Where P_{hm} is the probability of absent favorite database for hot group in memory buffers (*miss*) and P_{cm} is the same for cold group. By using (15) and (16) in (14), $\overline{x}$ can be calculated as follows:

$$\begin{aligned} \overline{x} &= P_h \overline{x}_h + P_c \overline{x}_c \\ &= P_h (P_{hm} T_{pd} + T_{pm}) \\ &+ P_c (P_{cm} T_{pd} + T_{pm}) \\ &= (P_h P_{hm} + P_c P_{cm}) T_{pd} + T_{pm} \end{aligned} \quad (17)$$

Let β_h be the probability of access to a hot database page and β_c be that for a cold database page. Assuming that memory buffers for hot and cold groups are shared, the probability of *miss* for them is as follows:

$$P_{hm} = (1 - \beta_h)^m \beta_h \quad (18)$$

$$P_{cm} = (1 - \beta_c)^m \beta_c \quad (19)$$

For splitting data into two groups, each subscriber record is split into two parts. Let R be the size of subscriber record, R_h be the size of its hot part and R_c be the size of its cold part. It is obvious that $R = R_c + R_h$.

Hot and cold tables are indexed separately. The number of records in hot and cold tables is alike but amount of information in the leaves of index tree are different. With the above descriptions and according to (10), β_h and β_c can be calculated as follows:

$$\beta_h = \frac{3}{4}\frac{P}{N.R_h} \quad (20)$$

$$\beta_c = \frac{3}{4}\frac{P}{N.R_c} \quad (21)$$

Finally, comparing with data flat case, access time to the database in data split denoted by W_{split} , is derived as follows:

$$W_{split} = \alpha_{split} T_{pd} + T_{pm} + W \frac{\frac{4}{3}(\gamma_r + 2\gamma_u)(\alpha_{split} T_{pd} + T_{pm})^2}{2[1 - (\gamma_r + 2\gamma_u)(\alpha_{split} T_{pd} + T_{pm})]} \quad (22)$$

Where

$$\alpha_{split} = \frac{3}{4} P_c (1 - \frac{3}{4}\frac{P}{N.R_c})^m \frac{P}{N.R_c} + \frac{3}{4} P_h (1 - \frac{3}{4}\frac{P}{N.R_h})^m \frac{P}{N.R_h} \quad (23)$$

In the next section, by applying typical values to the parameters in the above equations, we will present the diagrams which show the effect of data split on the access time to database. The diagrams also show the effect of data split ratio, the number of subscribers, the number of memory buffers and the size of disk page on the access time.

5. Numerical Example

In this section of the paper, we evaluate the result of the used method with a numerical example with practical network conditions. We have used a simple model from [7] and [9] for evaluating request arrival rate to the database. If user density is $\rho(user/km^2)$, their average velocity is $\upsilon(km/h)$, their moving directions are uniformly distributed in $[0, 2\pi]$ and the boundary length registration area for database is $L(km)$, γ_u and γ_r that are used in (3) are given by:

$$\gamma_u = \frac{\rho \upsilon L}{3600\pi} \quad (24)$$

$$\gamma_r = \frac{\rho r A}{3600} \quad (25)$$

Where A is the registration area and r is called originating rate per subscriber per hour. For simplification, we assume the area is a square that its boundary is $L(km)$ and its area is $L^2/16$. The higher user density or the wider registration area causes the higher rate of database access. On the other hand, the number of subscribers depends on both of them, too. Let registration area be a square, the number of subscribers denoted by N can be calculated as

$$N = \frac{\rho L^2}{16} \quad (26)$$

To get the numerical results, we primarily use the same values as those in [7]. The results are shown in Table 1.

Table 1. Typical values of network parameters relevant to database.

ρ	L	υ	r	N	γ_u	γ_r
390	30.3	5.6	1.4	22400	5.9	8.7

Typical values for other effective parameters in (12) and (22) are shown in Table 2. Using these values, service time and also access time are calculated in two cases, split and flat.

It is observed from Table 3 that in defined conditions, service and access time values with split method are approximately 1/4 of those with flat method. In data grouping into hot and cold parts, split ratio is an effective factor on access time. Note that modifying split ratio changes the access probability to each part. The most suitable split is when the size of hot part is smaller than cold part and access probability to it is higher. The larger hot part causes the higher access probability to it, but it has an undesired effect on the access time. For evaluating the split ratio changes and their effect on the access time, let hot part grows from a small value to a maximum value. In a logical growing at the first, parts of subscriber's record are added to hot part that have more effect on access probability while adding remaining parts have less effect on access probability. So, we can say that growth of records in the hot part closes the access probability to 1 in an exponential way as follows:

$$P_h = 1 - e^{-\alpha R_h} \tag{27}$$

Where $e^{-\alpha R} \approx 1$. Depending on α, P_h is closed to 1 with different speeds. As a typical value, α can be defined as 0.02. The values in Table 2 should satisfy (27), so 0.02 is a suitable value. The following diagrams show the effect of subscriber's record split ratio on access time. In these diagrams, there is a minimum point. It means that there is a split ratio for minimizing the database access time in every condition, also there is a split ratio to maximize the database access time.

Table 2. Typical values for effective parameters in access time.

P_h	P_c	T_{pd}	T_{pm}	R	R_h	P	m
0.8	0.2	0.1	10^{-8}	10^3	100	8K	2000

Table 3. Service and access time values in split and flat cases.

	α	$\bar{x}$	W
Flat	5.8×10^{-5}	15.9μsec	20.9μsec
Split	4.2×10^{-5}	4.2μsec	4.5μsec

In the curves of Figure 3, the database access time values are 3.51 and 21.31 for flat case. It may be readily seen that access time value approaches to its value for flat case. Note that access time value become more than flat case value for some split ratios, so, some of the split ratios force to fail our method. This situation shows that the data split method doesn't optimize with any split ratio and our favorite split ratio has a validation domain.

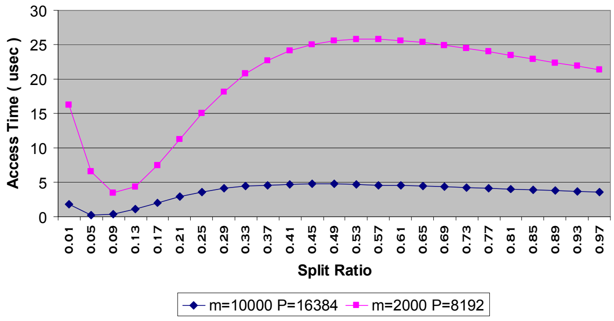

Figure 3. Database access time versus split ratio.

6. Simulation Results

In this section, we will evaluate theoretical results with our assumptions in analytical model by simulation of network conditions. The simulation is based on the database model in Figure 1. In this simulation it is assumed that:

1. The rate of arrival requests is according to Poisson distribution.
2. Subscriber's database has been created based on [1].
3. Database implementation is performed by MySQL database engine based on the Linux operating system and database index structure is clustered-B-tree.
4. Network parameters have the values written in Table 4.
5. Database parameters have values mentioned in Table 5.

Table 4. Numerical assumptions for network simulation.

ρ	L	υ	r	N	γ_u	γ_r
800	30.3	5.6	1.4	45950	12.1	17.8

Table 5. Numerical assumptions for database simulation.

R	α	P	m
10^3	0.02	16K	1280

Figure 4 shows the access time of simulated database, where split ratio starts from 0 and approaches to 1.

From Figure 4, it can be noticed that simulation results obey theoretical results, it can also seen the effect of split ratio on the database access time. From the figure, we observe as the split ratio approaches to 1, at first, access time decreases as well. After a minimum point it increases and finally approaches to a value while passing a maximum point.

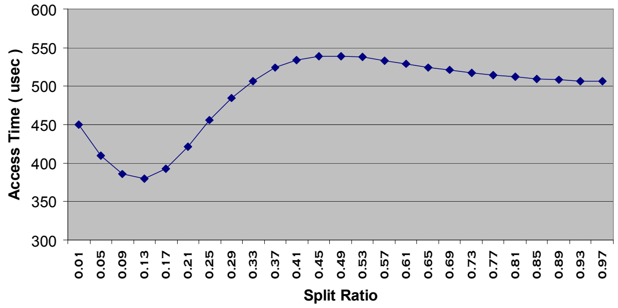

Figure 4. The access time of the simulated database versus split ratio.

7. Conclusions

One of the most important subjects in mobile systems in future is user service time. Major part of this time is consumed for database access. Considering future, we have presented a method that optimizes database access time. This method uses data grouping concept.

In this article, by analytical model and also simulation, we proved that the data split method optimizes database access time. We also noticed that the performance of our method has the limitations which depend on the split ratio values. With a split ratio out of a valid domain, the data split method has a reverse effect on the database access time. Data split is an idea for increasing database performance in mobile networks and research in this field can be continued. This method can be also applied on other databases and its performance depends on the quality of data split. The more differences of access rate to splitting parts of data, the higher quality of data split.

In this article, data split was performed in two groups. Data split in various groups based on access rate will be an interesting subject for future researches.

References

[1] Effelsberg W. and Haeder T., "Principles of Database Buffer Management," *ACM Transaction on Database Systems*, vol. 9, no. 4, pp. 560-595, December 1984.

[2] ETSI GSM Technical Specification, "Digital Cellular Telecommunications System (Phase2+) (GSM); Mobile Application Part (MAP) Specification (GSM 09.02)," *TS 100 974*, V7.5.1, 2000.

[3] ETSI GSM Technical Specification, "Digital Cellular Telecommunications System (Phase2+) (GSM); Organization of Subscriber Data (GSM 03.08)," *TS 100 526*, V7.2.0, 1999.

[4] Gokmen Gok H. and Ulusoy O., "Transmission of Continuous Query Results in Mobile Computing Systems," *Information Sciences Applications: An International Journal*, vol. 125, no. 1-4, pp. 37-63, June 2000.

[5] Gross D. and Harris C. M., *Fundamentals of Queuing Theory*, John Wiley & Sons, 1985.

[6] Mohan S. and Jain R., "Two User Location Strategies for Personal Communications Services," *IEEE Personal Communications*, First Quarter, vol. 1, no. 1, pp. 42-50, 1994.

[7] Mao Z. and Douligeris C., "A Distributed Database Architecture for Global Roaming in Next-Generation Mobile Networks," *IEEE/ACM Transactions on Networking*, vol. 12, no. 1, February 2004.

[8] O'Neil E. J. and O'Neil P. E., "The LRU-K Page Replacement Algorithm For Database Disk Buffering," *in Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data*, NY, USA, pp. 297-306, 1993.

[9] Qiu X. and Li V. O. K., "Performance Analysis of PCS Mobility Management Database System," *in Proceedings of Int. Conf. on Computer Communications and Networks (IC3N)*, USA, pp. 434-441, September 1995.

[10] Zawodny J. and Balling D. J., *High Performance MySQL, Optimization, Backups, Replication, Load Balancing & More*, O`Reilly, April 2004.

Mehdi Samie received his BS in computer engineering from Isfahan University of Technology (IUT), Isfahan, Iran, in 1996, and MS in computer architecture from Iran University of Science and Technology (IUST), Tehran, Iran in 2001. Since 1999, he is teaching at the Department of Computer Engineering. He joined the Iran Telecommunications Research Center (ITRC) in September of 2001, where he is currently pursuing research on communication protocols and subsystems for NGN and IMS networks as a research assistant. His research interest includes signaling protocols, mobile network protocols, mobile IP, digital transmission, and database in telecommunication networks.

Hassan Moradi received his BS degree in electrical engineering from Shiraz University, Shiraz, Iran in 1998, and MS in communication systems from K. N. Toosi University of Technology, Tehran, Iran in 2001. He joined the Iran Telecommunications Research Center (ITRC) as an assistant researcher of wireless communications in 2001. Currently, he is a member of technical staff of ITRC. He has authored several papers in the wireless communications field especially about access network of CDMA-based systems. Currently, he works and manages the implementation of Switching Type Approval Platform (STAP) project at ITRC. His research interest includes mobile communications network, radio transmission, and signaling protocols.

Nafiseh Saberi received the BS degree in computer engineering from Azad University of Shiraz, Shiraz, Iran, in 2001. She joined the Satellite Group at Iran Telecommunications Research Center (ITRC) as an assistant researcher in June of 2003. Currently, she is pursuing research on signaling protocols for the implementation of Switching Type Approval Platform (STAP) project with Switch Group at ITRC. Her research interest includes communication protocols, computer networks, genetic algorithms, and database in telecommunication networks.