

Consistency preservation techniques for Location Register System in Mobile Networks

Jang-Hwan Kim*

Professor, Dept. of Media S/W, Sungkyul University Korea
jhkim@sungkyul.ac.kr

Abstract

A database called Home Location Register(HLR) plays a major role in location management in mobile cellular networks. The objectives of this paper are to identify the problems of the current HLR system through rigorous analysis, to suggest solutions to them. The current HLR backup method is a process of simply writing the changed memory SLD block to disk, which has a problem in maintaining database consistency. Since information change and backup are performed separately by separate processes, there is a risk of information inconsistency when an error restart occurs. To solve this problem, a transaction concept was introduced for subscriber-related operation functions and a recovery method through logging and checkpointing was introduced. The subscriber related functions of tasks terminated normally by the suggested process are recovered with consistency even after system restarts. Performance is also not affected seriously because disk tasks for log occur with only subscriber related functions.

Keywords: *Consistency , Recovery, Transaction, Backup*

1. Introduction

Most fields like mobile communication embedded system, home network, and telematics require the management of location information essentially. Thus HLR system will be used in IoT network and satellite mobile communication as expanding functions of the system. In addition to that, there are the requirement of dynamic scheme evolution according to the increase of supplementary services, subscriber migration among databases, and assistance for special supplementary services. HLR database system, therefore, needs to reflect those features[1][2]. Mobile communication system is expected to be evolved into the personal mobility services from the terminal mobility services. It is also evolving to take charge of international roaming services and information security management.

In this paper, I suggest the techniques of consistency preservation for HLR database system and S/W structures applied practically to develop better system, which are based on the need of the problems from the results of analyzing the current HLR database system[3][4].

The paper is organized as follow. I first describe the current S/W structures in HLR system in section 2.

Second, the proposed S/W structures for consistency preservation in HLR database system are represented in section 3. In section 4, I finally bring my research to conclusion.

2. The software structure of HLR system

HLR system connects MSC/VLR(Mobile Switching Center, Visitor Location Register). Its main function is to process it in real time to support call processing through location information management of subscribers, and manage or support supplementary service information for each subscriber, which is currently used in Cellular Network[5][6][7].

The application service entities are a part of MAP operation in HLR software and is related to database system to update or inquire location information. It are implemented on the basis of MAP from KMAP(Korean-MAP)[8][9] or IS(Industry Standard)-41(A)[10] and makes each operation a process. ASE and MAP are interchangeably used in general. The aims for operation and maintenance part can largely have two sides. First, one is the functional side to collect, analyze, and manage the information as statistics, status, and failure required for HLR system operation. Second, another is the user-friendly side to operate more conveniently. The transaction capabilities application part is the protocol layer connecting between Application Service Entities and Signalling Connection Control Part in common channel signaling system protocol. This part is related to protocol for communicating with MSC/VLR without requiring the operation related to database system. The messages delivered from MSC/VLR are, via the message transfer of SINAP software or signalling connection control part, transferred to the transaction capabilities application part, in which each message according to the type of the messages are delivered to the component process of the corresponding application service. The related works through database engine process in the component process of application service are executed and the results are oppositely transferred in the process of delivering messages. The transaction capabilities application part does not take the job to access database. ATIP process is currently between transaction capabilities application part and the component process of application service.

3. The proposed consistency preservation techniques for HLR database system

3.1 The proposed database system architecture

In case of database's crash due to system error, logging mainly is used to make database the recent state with consistency. Transaction concept should be introduced for the related operation of subscribers in HLR database system, for which database should keep consistency. Since the existing method can not keep such transaction consistency, transaction or logging method needs to be introduced in the related tasks of subscribers.

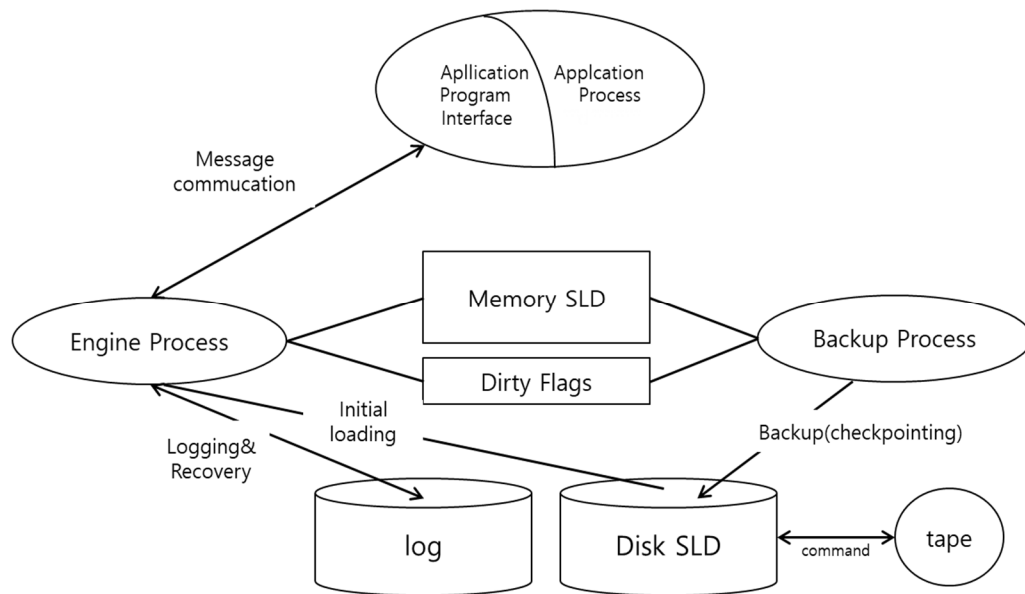


Figure 1. Proposed Location Register database system architecture

The instances of procedures performed by transaction of each message are as follows.

There are additional, deleting, and altering message of subscribers or subscriber information.

The additional message of subscribers

- 1) make sure that the terminals of relevant subscribers exist in the table of victim-subscribers of theft.
- 2) add the subscriber by calling insertion function
- 3) increase the number of subscribers in prefix table by calling update function

The deleting message of subscribers

- 1) delete subscribers in the tables of subscribers by calling deletion function
- 2) decrease the number of subscribers in prefix table by calling update function
- 3) delete the number if the terminal number of the relevant subscriber exists in the table of victim-subscribers of theft.

The altering message of subscriber information

- 1) change the information of relevant subscribers by calling update function
- 2) add or delete the attributes in the tables of victim-subscribers of theft if the attributes of victim-subscribers of theft are changed.

As seeing the above, a message consists of several database tasks. The job takes atomic nature. So it should be implemented that the prior state which did not deal with messages could be recovered if errors happen in the process of message processing. If the changes of database happens as insertion, deletion, and update, they are registered as log by logging management part. The query processing takes the responsibility for transaction abortion of database and logging and recovery management part take the responsibility for the restoration in crash.

If errors occur in the deleting process of subscribers, withdrawal immediately happens without any other recovery operations. But the occurrence of errors in the change of prefix table or the deleting process in the

tables of victim-subscribers of theft makes withdrawal after executing addition of subscribers and the recovery operation of reallocation in prefix table. Each updating task is registered as log and prepare for crash. The existing query processing performed the functions by calling db-selectTup() and db-insertTup() for each message of retrieval and insertion. So, additional messages of newly additional subscribers, deleting messages of subscribers, and the changing messages of subscriber information should be implemented by adding db-insertSub(), db-deleteSub(), and db-updateSub() as the same way of the above process.

The existing HLR database system supports subscriber tables, prefix tables, and victim-subscriber tables of theft. Table management controls the three tables as one dimensional array in shared SLD memory of UNIX. The insertion and deletion of tuples are performed by using USED field of the first byte of each tuple and the free list of catalog information. There are two kinds of log as physical log and logical log. Physical log stores how some content is changed in some address and logical log stores what operation has existed. As only the related operation of subscribers needs log in HLR database system, it can be implemented by logical logging method. Next Figure 2 shows log's structures and types used in HLR database system.

log's structures: **⟨TransactionID, OperationID, TupleContent⟩**
types of Operation: **START, COMMIT, INSERT, DELETE, UPDATE**

Figure 2. log's structures and types

Log structure consists of TransactionID to represent what transaction, OperationID to represent what database operation, and TupleContent to store the the tuples used on the operation. Database operation comprise START and COMMIT to notify the beginning and end of transaction. It also comprise INSERT, DELETE, and UPDATE of operations database alterations occur.

The reason why the contents of prior tuples are not saved is that not UNDO function but only REDO function needs owing to the characteristics of main memory database system. That is, as UNDO is naturally carried out due to the feature of volatile main memory, prior tuple contents does not have to be stored. On the other hand, because REDO should be executed by using the tuple contents subsequent to alteration in COMMIT transaction, the contents should be saved in safe storage of disk. The following Figure 3 shows an example of log owing to dealing with deletion transaction of subscribers. As the Figure shows, the log comprises four or five entries according to the existence of victim-subscribers of theft. In system crash, the use of this log can bring the reexecution of the deletion function of relevant subscribers.

⟨1001,START,null⟩
⟨1001,DELETE,(01054551644,9F48001E,.....)⟩
⟨1001,UPDATE,(0105455,.....)⟩
⟨1001,DELETE,(9F48001E,.....)⟩
⟨1001,COMMIT,null⟩

Figure 3. log of deletion transaction of subscribers

The work related with all subscribers can recognize the results from log. Therefore, operating system long time makes so many logs that it takes much time to restore from system error. The problem can be solved by making checkpoint in backup process. That is, if checkpoint, transferring SLD memory to SLD disk, is completed, prior logs are deleted in disk because the recent database is stored. Such checkpoint tasks are

executed at nights or at holidays. In result, as transactions, when making checkpoint, do not occur, it is possible to execute database backup with consistency of transaction naturally.

Logs about operation function related with subscribers remain by using the function of logging management in query processing. Such logs are used for recovery in system crash. So recovery is achieved after loading SLD disk onto SLD memory by engine process in reoperation of system. That is, the recovery function is performed by calling the function of recovery management part in initializing processing part of engine process. The process executing recovery is as follows. First, find transaction with COMMIT, searching the logs stored in disk from the earliest point to the recent checkpoint in order. Next, the found transactions execute REDO function as the order of stored logs. On the other hand, by ignoring transactions, not having COMMIT logs, UNDO is naturally made to happen. The recovery management part offers only `initial_recovery()` function to initializing processing part. That is, initializing processing part executes recovery by calling the `initial_recovery()` function after loading database. The function searches transactions with COMMIT and execute REDO function in loaded main memory database. From the results, normal services can keep going by composing database with transaction consistency.

3.2 The improvement for the problem with inconsistency

The current search request message, insertion request message execute a database task respectively. But the above additional messages of subscribers should execute several database tasks in engine process. So, after engine process receives additional messages of subscribers, it deals with a transaction of functions which are search to prevent subscriber redundancy or prefix number redundancy, addition of subscriber tuples, and subscriber number alteration of prefix tables with relevant subscribers. After success or failure of the transactions is determined, the reason consisting of response messages is delivered into application process in case of failure. Messages about deletion of subscribers and registration and deletion of supplementary service should be implemented as similar way as well as additional request messages of subscribers. Such transactions are separated into different messages each other and transferred from operation and maintenance system or application process of customer management center into engine process, by which the functions are executed and only the results are delivered to application process again.

The alteration of SLD memory is logged in disk including the starting and ending points of those transactions. The log is used for recovery in case of restart due to system error. By considering too many logs owing to long-time operation, checkpoint tasks to copy the contents of SLD memory into disk are executed in spare time of system. The checkpoint tasks are executed in night or holiday, during which operations related with subscribers do not happen. This naturally solves the problem which transactions should be aborted in a point of checkpoint time.

When HLR system restarts owing to errors as power-off, recovery is achieved by very simple process. As the whole contents of main memory database disappear, different from database based on disk, extra UNDO function does not need. Therefore, REDO task can be achieved by applying logs of transactions with COMMIT sequentially after the point of time to SLD disk stored in the point of recent checkpoint time.

4. Conclusion

In this paper, I suggested effective S/W architecture based on the problems found in HLR system operated in current mobile communication network. As the current backup method registers just altered SLD memory blocks in disk, I indicated that it can not keep consistency. To solve the problem, it needs to introduce transaction concept for operation function related with subscribers and recovery method through logging and checkpoint. The alteration of location information due to location registration does not bring quick backup.

On the other hand, operation function related with subscribers makes transactions of functions which are split as addition and deletion of subscribers and registration and deletion of supplementary. These improvements will help HLR system development in IoT network or satellite communication in the future. The most important requirements in mobile commerce based on evolution into advanced service in the future will be security functions. The minimal attributes to be changed in relation of current HLR DB Scheme for such function are as follows. To perform mutual authentication protocol or establish session key, the attributes, representing the result values or session key from mutual authentication algorithm and random number used in the algorithm, should be added. Before roaming heterogeneous networks frequently with evolution into advanced service in the future, effective security function development about session key or relevant authentication variable is essential.

References

- [1] Michael Freitag, M.Sc., Prof. Alfons Kemper, Ph.D. Garching Michael Freitag, M.Sc., Prof. Alfons Kemper, Ph.D. Garching, "Seminar: Implementation Techniques for Main Memory Database Systems", June, 2018.
- [2] F. Faerber, A. Kemper, P. Å Larson, J. Levandoski, T. Neumann, and A. Pavlo. *Main Memory Database Systems. Foundations and Trends in Databases*, vol. 8, no. 1-2, pp. 1–130, 2016.
- [3] M. Min, "Experiments of Search Query Performance for SQL-Based Open Source Databases", *International Journal of Internet, Broadcasting and Communication*, Vol. 10, No.2, pp. 31-32, 2018. DOI: <https://doi.org/10.7236/IJIBC.2018.10.2.6>
- [4] R. Elmasri, and S. B. Navathe, *Fundamentals of Database Systems, 7th Edition*, Pearson, 2016.
- [5] Sung-Moon Michael Yang, *Modern Digital Radio Communication Signals and Systems*, Springer, 2019. DOI: <https://doi.org/10.1007/978-3-319-71568-1>
- [6] 3GPP, *Technical Specification Group Core Network and Terminals*, TS 29.011, 2014.09.
- [7] Y. Xiao, Y. Pan, and J. Li, " Design and Analysis of Location Management for 3G Cellular Networks," *IEEE Transactions on Parallel and Distributed Systems*, Vol. 15, No. 4, pp. 339-349, April 2004.
- [8] TTA, *MSC/VLR-HLR/AC MAP*, 1997.7,
- [9] TTA, *ANSI-41/WIN - International implementation of Wireless Telecommunications Systems Compliant With ANSI/TIA/EIA-41*, 2002.10.28
- [10] TIA/EIA, *IS-41(A)*, Initial Version. Jan. 1991.