

Available online at <http://www.mecspress.net/ijwmt>

Crash Recovery for Embedded Main Memory Database

Yi-bing Li^{a,*1}, Shuang Wang^{a,*2}, Fang Ye^a

^a College of Information & Communication Engineering, Harbin Engineering University, Harbin, China

Abstract

In the last years, the deployment of embedded real-time communication systems has increased dramatically. At the same time, the amount of data that needs to be managed by embedded real-time main memory databases is increasing, thus requiring an efficient data management. However, system crash will cause data loss in main memory, which will seriously affect the normal operation of the entire communication system. This paper introduces an algorithm of system crash recovery applied in main memory database of embedded real-time communication system. This paper expatiates the software architecture of SDR base station, the cause of crash occurring, and proposes the flash recovery algorithm using for system recovery. A main memory database using this algorithm has been applied in a real communication system—multimode SDR base station communication system.

Index Terms: Crash Recovery; Embedded Main Memory Database; System Crash; Multimode SDR Base Station

© 2012 Published by MECS Publisher. Selection and/or peer review under responsibility of the Research Association of Modern Education and Computer Science

1. Introduction

Memory database history can be traced back to 1969 when IMS was developed by IBM. And main memory database has aroused stronger and stronger concern by database researchers since the early 1980's. The most prominent feature of main memory database is that the working copy of the database is resident in memory but the data is very weak, so recovery in main memory database is extremely important. Furthermore, recovery is the only part related to I/O process so that the performance of the recovery is the key to the performance of the entire system [1].

Main memory database is very different from the traditional database system because of the data structure, transaction processing methods and optimization, concurrent control and recovery, etc [2]. It is necessary to design new algorithms and data structures for main memory database [3-5] which are different from disk database system. A new crash recovery algorithm for main memory database is proposed in this paper, and it is particularly effective for system crash recovery. The algorithm combines the features of flash saving and

This work is supported by shipbuilding Defense Advance Research Program (10J3.1.6) and Research on technological innovation Harbin Special Fund (2008RFQXG0205).

* Corresponding author:

E-mail address: ^{*1} liyibing@hrbeu.edu.cn; ^{*2} wangshuang41@hrbeu.edu.cn

introduces flash writing log to ensure data consistency real-timely. In addition, the algorithm is suitable for multimode SDR base station communication system.

2. Crash Recovery for Embedded Main Memory Database

2.1. SDR base Station Communication System

SDR base station can be upgraded and evolved just by adding or replacing the base band processor board, the rest of the base station can be realized through software configuration to another format of a standard smooth evolution of the technology. The software architecture of SDR base station system is as shown in Figure 1.

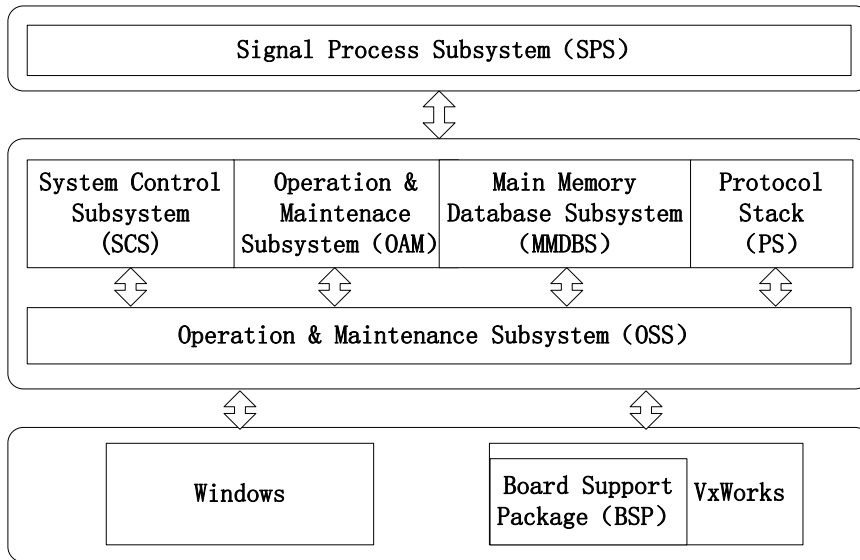


Fig. 1. Software Architecture of SDR Base Station System

SPS is to complete the business functions of the system, such as GSM service, LTE service and CDMA service. The responsibility of SCS is power control of the system. OAM manages the entire system initialization at power, and resource allocation and maintenance at run-time. PS includes network protocol layers. MMDBS is the core of the system, and it is responsible for data management and maintenance of the entire system, and provides data interfaces to other subsystems, and receives configuration data from the background by OAM. All the above subsystems are required to obtain the supports from OSS and OS.

2.2. Crash in Main Memory Database

The crash of main memory database can be divided into three categories [2], the crash in transaction, media crash and system crash. When the crash in transaction happens, the database may be in incorrect state, and the internal affairs don't achieve the desired end. While media crash means external memory crash, such as disk crash, disk head impact and transient electromagnetic interference. This kind of crash will affect the whole database or part of it, but the possibility of such crash is small. System crash contains all kinds of events which can cause the system to stop running, and the system must restart. This kind of crash includes some certain

types of hardware errors, OS crash, DBMS code error, and sudden power down. Such crash will influence all the running affairs, data in main memory will loss, and all the running business will unusually terminate. When system crash happens, it is difficult to ensure data consistency. Some child tables may have been sent to the physical database, but the corresponding father tables have not. It is necessary to clear all changes made by this transaction to ensure data consistency.

As the different design between main memory database and traditional database, there are great differences in crash causes and recovery methods, which is mainly in the following two aspects:

- 1) The focus in main memory database is to recover the memory database rather than the disk database. It is generally considered that the key to the system data consistency is good recovery after system crash.
- 2) Recovery performance becomes more and more critical to the overall performance of the whole system. And some traditional methods are no longer suitable, such as REDO/UNDO logging, checkpoint synchronization, and recovery when in the “resting state”.

2.3. Crash Recovery Algorithm for Main Memory Database

The embedded main memory database is also a relational database, and system tables are relational tables. There may be parent-child constraints between tables; the database must ensure the consistency of parent-child tables' data at any time. Therefore, a new method of crash recovery called flash recovery method is proposed in this paper.

For multimode SDR base station system, each product has its own relational tables. Those which are co-dependent parent tables constitute a public statement, such as radio resource table. And child tables and tables doesn't exist parent-child relationships constitute their own product tables. For example, in the GL (GSM and LTE) dual-mode SDR base station, there will be public tables, GSM product tables, and LTE product tables. The flash recovery process will start when changes in the memory happens. Tables will be saved in flash corresponding to these three libraries: public library, GSM product library, and LTE library.

In order to distinguish the save path of each library and avoid the situation that some tables write failure while some tables write successfully, leading to the base station can't function properly, each path has two file writing directory (A and B). The file writing directory is shown in Figure 2.

```

|Public \
    |Directory A\
    |Directory B\
|GSM product\
    |Directory A\
    |Directory B\
|LTE product\
    |Directory A\
    |Directory B\

```

Fig. 2. File Writing Directory

Flash writing operation of the public library is to write tables of the public library to the subdirectory A and B under the public main directory respectively. And flash writing operation of the certain product library is to write tables of the corresponding product library to the subdirectory A and B under the corresponding product main directory respectively. A flash writing logo is introduced to identify the progress. So if a sudden system

crash happens when the system is running, such as power down, the system will go to the correct flash directory to load data in order to ensure the data integrity after restart. It is the key to modify the flash writing logo after all tables have written over in directory A or B. The flash writing process is shown in Figure 3.

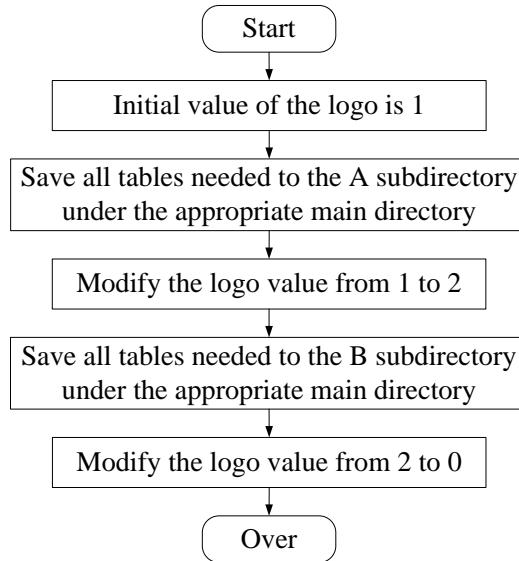


Fig. 3. Flash Loading Process

Then the system will load data to the memory from the correct flash directory after system crash, the process is shown in Figure 4.

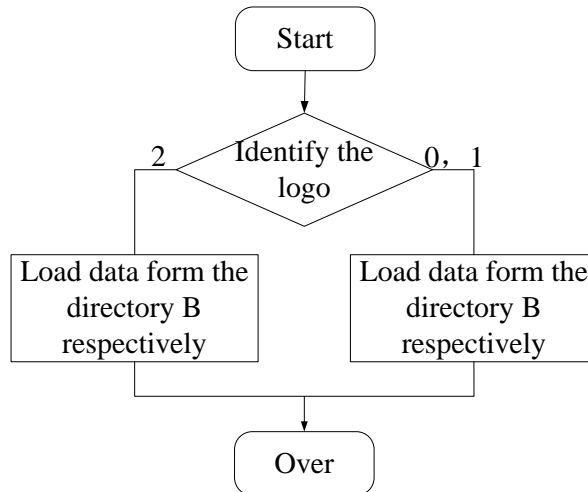


Fig. 4. Flash Loading Process

The data under directory B is complete when the logo symbol 0 or 1, and loading data from B is correct. And the data under A is complete when the logo symbol 2, so loading data from A at this situation.

3. Conclusion

As the serious destructiveness of system crash for main memory database, the system crash recovery algorithm for main memory database system is particularly important. The crash recovery algorithm proposed in this paper combines the characteristics of main memory database and flash memory for system crash, and it is very capable for multimode SDR base station system.

References

- [1] Ping Liang, Yunsheng Liu, "A checkpointing strategy and redo point strategy for embedded real-time main memory databases crash recovery," *2009 WRI World Congress on Computer Science and Information Engineering*, vol.2, pp.696-700, March 2009
- [2] Tang Yanjun, Luo Wen-hua, "A model of crash recovery in main memory database," *2010 International Conference On Computer Design And Applications (ICCD)*, vol. 5, pp. 205-207, June 2010.
- [3] Jiang Hao, Zhang Li, "Design of Distributed Main Memory Database Based on an Improved P2P Routing Algorithm," *2009 1st International Conference on Information Science and Engineering(ICISE)*, pp.190-193, Dec 2009.
- [4] Qin Xiangpai, Zhou Xiaoyun, "Performance study on logging to another main memory database," *2010 4th International Conference on New Trends in Information Science and Service Science(NISS)*, pp. 36-40, May 2010.
- [5] Qin Xiangpai, Xiao Yanqin, Cao Wei, Wang Shan,"A Parallel Recovery Scheme for Update Intense Main Memory Database Systems," *PDCAT Ninth International Conference on Parallel and Distributed Computing, Applications and Technologies*, pp. 509-516, Dec 2008.