CHAPTER 1

MOBILE DATABASE SYSTEM

1.1 INTRODUCTION

This chapter presents an intuitive introduction to the architecture and basic components of a *mobile database system (MDS)*. Such an introduction is very helpful in preparing our readers to understand the complete architecture, its functionality, and research and development issues. The introduction briefly explains the objectives of MDS, describes the data organization, discusses the role of each component, explains how transactions are processed, describes the role of concurrency control mechanisms, and discusses database discovery issues.

1.1.1 Mobile Database System Architecture

An MDS is a distributed multidatabase client/server system built on a PCS (personal communication system) [1] or a GSM (global system for mobile communication) platform [2–4]. It provides complete database functionality to any user—mobile or stationary. End users, therefore, are unable to differentiate whether they are using an MDS or a conventional database system. A user can connect to the database from anywhere (driving, traveling on a plane, enjoying a roller-coaster ride, etc.) without having to go through the web. The connectivity between the user and the database system is established either through a cellular network or through satellite communication. There are some differences in GSM and PCS architectures; as a result, the architecture of MDS built on these platforms may differ slightly; however, it does not affect MDS functionality [5, 6].

Fundamentals of Pervasive Information Management Systems, Second Edition. Vijay Kumar.

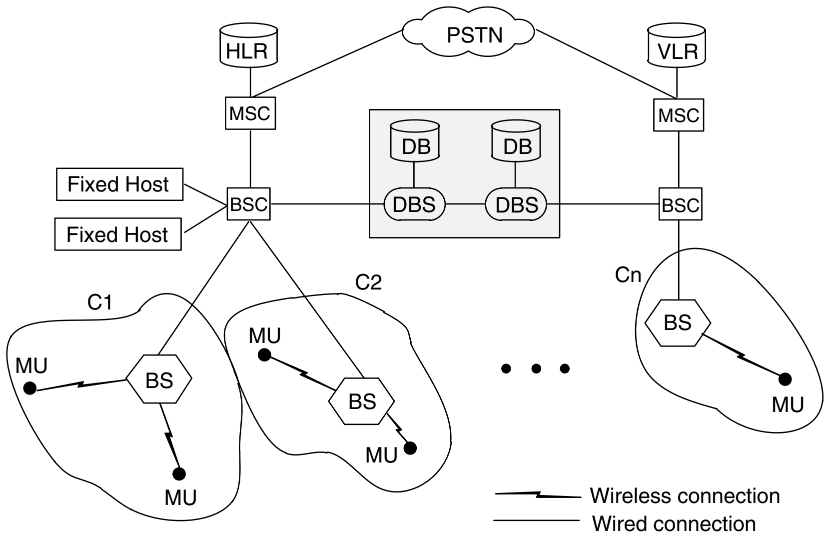

Figure 1.1 Reference architecture of a mobile database system.

Database functionality is provided by a set of DBSs (database servers) that are incorporated without affecting any aspect of the generic cellular network. A reference architecture of MDS is illustrated in Figure 1.1.

The architectures of GSM and PCS are discussed in detail in Chapter 2. The components of these systems are special-purpose computers (switches). The reference architecture of MDS is, therefore, described in terms of components of GSM/PCS such as *base station (BS)*, *fixed hosts (FHs)*, *base station controller (BSC)*, *mobile switching center (MSC)*, *home location register (HLR)*, and *visitor location register (VLR)*. These components are described in detail in Chapter 2. We refer to some of these components as workstations, PCs, PDAs, laptops, and so on, to better understand the architecture and functionality of MDS.

The reference architecture of our MDS that is discussed here is based on a GSM platform. In MDS, a set of general-purpose computers are interconnected through a high-speed wired network and categorized into *fixed hosts (FHs)* and *base stations (BSs)* or *mobile support stations (MSSs)*. FHs are not fitted with transceivers so they do not communicate with mobile units (laptop with wireless connectivity, PDAs, IPad, etc.) One or more BSs are connected with a *BSC* that is also referred to as the *cell site controller* [1, 4]. It coordinates the operation of BSs using its own stored software programs when commanded by the MSC. To coordinate with DBSs, some additional simple data processing capabilities are incorporated in BSs. Unrestricted mobility in PCS and GSM is supported by a wireless link between BS and mobile units. These mobile gadgets are referred to as *mobile hosts (MHs)* or *mobile units (MUs)* [1, 4]. BSs are fitted with transceivers and communicate with MUs through wireless channels and link wired and wireless parts of the MDS. Each BS serves one cell that is a well-defined geographical area. The size of a cell depends on the power of its BS. In reality, a high-power BS is not used because of a number of factors; rather, a number of low-power BSs are deployed for managing movement of MUs in various cells.

Database servers (DBSs) can be installed (a) inside BSs or (b) inside FHs. There are, however, a number of problems with this setup. BSs or FHs are switches and they have specific switching tasks to perform; incorporating database capability in them would interfere in their switching tasks. Besides, the size of a BS varies and inhibits incorporating complete database functionalities in smaller BSs, and the entire architecture of a BS (hardware and software) may have to be revised. Such a major change in a cellular platform would be unacceptable from a mobile communication viewpoint. In addition to this, the setup will not be modular and scalable because any change or enhancement in database components will interfere with data and voice communication. For these reasons, DBSs are connected to the system through a wired network as separate nodes as shown in Figure 1.1. Each DBS can be reached by any BS or FH, and new DBSs can be connected and old ones can be taken out from the network without affecting mobile communication. The set of MSCs and PSTN connects the MDS to the outside world.

A DBS communicates with MUs only through BSs. The duration a mobile user (MH or MU) remains connected to DBSs (through BSs) depends on the workload. On the average, it is actively connected for 2 to 4 hours during a day. At other times, when it is not engaged in transaction processing, it must save the battery power. To conserve the power, a mobile unit can be switched to stay in (a) powered off mode (not actively listening to the BS) or (b) an idle mode (doze mode—not communicating but continuously listening to the BS) or (c) an active mode (communicating with other parties, processing data, etc.) The unit can move from one cell to another in any of these modes. For example, a driver can switch its cell phone to save battery power while driving and may cross many cell boundaries. The total coverage area of an MH is the sum of all cell areas, that is $\{C_1 + C_2 + \cdots + C_n\}$ where C_i indicates a cell. As discussed in Chapter 2, the mobile unit encounters *handoff* only in active mode.

1.1.2 Data Distribution

The MDS has multiple DBSs and a database can be (a) partially distributed or (b) partitioned or (c) fully replicated [7–9]. However, in MDS, data distribution is architecture specific. An MDS can be either a federated or a multidatabase system; as a result, data distribution must follow the constraints of *location-dependent*, *location-aware*, and *location-free* data [10]. Under such constraints, unlike conventional data distribution, a data item can be replicated only at a certain DBS. For example, it does not make sense to duplicate or replicate the tax data of Kansas City to a DBS that serves the Dallas area. This implies that a global database schema may not be feasible in MDS. However, we will see later that under some situations, a full database replication is possible and highly useful. Generally, partial data replication is always possible.

1.1.3 Transaction Processing

A user initiates a transaction either at an MU or at a DBS. Since MDS is a distributed database system, the execution of a transaction follows a distributed transaction processing paradigm [11–13]. If the necessary data are available at the place of origin (mobile unit or DBS), then it either can be processed in its entirety as a single transaction or may be split into multiple *sub-transactions* and processed at multiple processing units (MU or DBS) for better throughput. On the other hand, if the place of origin does not have the entire data for the transaction, then it has to split into *sub-transactions* and is processed in a distributed manner. The process and mechanism of the transaction split, the mode of distributed processing, and so on, will be discussed in later chapters.

1.1.4 Concurrency Control Mechanism

Similar to conventional database systems, MDS also requires conventional concurrency control mechanisms to serialize the execution of concurrent or parallel transactions. There are a large number of serialization mechanisms developed for database systems such as two-phase locking, timestamping, multiversioning, and so on, but because of the unique requirements of MDS, they may not be satisfactory [14]. Conventional serialization mechanisms are process and communication intensive. MDS does not have a lot of *channels* and power to support the volume of communication it requires; as a result, new energy-efficient serialization mechanisms need to be developed or conventional schemes need to be revisited to make them energy-efficient. To provide better response and throughput, MDS needs serialization techniques that have (a) independent decision-making capabilities and (b) minimum locking and unlocking overheads. Later chapters explain the requirements and working of such concurrency control mechanisms and develops a few.

1.1.5 MDS Recovery

MDS has many unique requirements such as management of location-dependent data, transaction processing during handoff, etc. These requirements make recovery complex, compared to conventional database recovery. The complexity begins from log management. The main problem is the management of the log generated by mobile units [15]. Where and how to store the log, how to make the log available for recovery, how and where to recover, and so on, are some of the difficult issues that need efficient and cost-effective solutions. In the later chapters, we discuss all these issues in detail as well as their theoretical foundations and recovery protocols.

1.2 SUMMARY

This chapter presented a brief introduction to a reference architecture of a mobile database system (MDS) and explained how transactions are processed. The objective was to develop readers intuition so that they can get a clear understanding of unique and complex problems and issues of MDS. Since the backbone of MDS is wireless and mobile disciplines, we begin coverage of these topics in the next chapters.

EXERCISES

1.1 Explain the unique properties of mobile database systems and identify the similarities and differences between conventional distributed database and mobile database systems.

1.2 What is the difference between wireless communication and mobile communication? Explain your answer and give real-world examples to illustrate the difference.

1.3 Identify the the set of components that use wireless channels and identify the set of components that are connected with wired network. Explain the roles they play in mobile database systems.

1.4 Discuss the role of DBS in managing the database. Do you think DBSs should be made a part of BSs (housed in BSs)? First explain if it is possible to make them as a part of BSs and then comment on the benefits of this architecture. Suggest another location for DBSs that works fine without affecting functionality of the cellular system.

1.5 Design a database partition and distribution scheme that will improve transaction processing throughput and response time. Explain how transactions will be processed under your distribution scheme. (*Note*: This is a research-type question suitable for a graduate class project.)

REFERENCES

1. R. Kuruppillai, M. Dontamsetti, and F. J. Cosentino, *Wireless PCS*, McGraw-Hill, New York, 1997.
2. D. Barbara, Mobile Computing and Databases—A Survey, *IEEE Transactions on Knowledge and Data Engineering*, Vol. 11, No. 1, January 1999, pp. 108–117.
3. T. Imielinksi and B. R. Badrinath, Wireless Mobile Computing: Challenges in Data Management, *Communications of the ACM*, Vol. 37, No. 10, October 1994, pp. 18–28.
4. M. Mouly and Marie-Bernadette Pautet, *The GSM System for Mobile Communications*, Cell & Sys Publications, Palaiseau, France, 1992.
5. E. Pitoura and G. Samaras, *Data Management for Mobile Computing*, Kluwer Academic Publishers, Hingham, MA, 1998.
6. E. Pitoura and B. Bhargava, Building Information Systems for Mobile Environments. In *Proceedings of the Third International Conference on Information and Knowledge Management*, Washington, D.C., November 1994, pp. 371–378.
7. D. Barbara and H. Garcia-Molina, Replicated Data Management in Mobile Environments: Anything New Under the Sun? In *Proceedings of the IFIP Conference on Applications in Parallel and Distributed Computing*, April 1994, pp. 237–246.

8. Y. Huang, P. Sistla, and O. Wolfson, Data Replication for Mobile Computers. In *Proceedings of the 1994 SIGMOD Conference*, May 1994, pp. 13–24.
9. D. Lee, W.-C. Lee, J. Xu, and B. Zheng, Data Management in Location-Dependent Information Services, *IEEE Pervasive Computing*, Vol. 1, No. 3, July–September 2002, pp. 65–72.
10. V. Kumar and M. Dunham, Defining Location Data Dependency, Transaction Mobility and Commitment, Technical Report 98-cse-1, Southern Methodist University, Feb. 98.
11. P. K. Chrysanthis, Transaction Processing in Mobile Computing Environment. In *IEEE Workshop on Advances in Parallel and Distributed Systems*, October 1993, pp. 77–82.
12. M. H. Dunham, A. Helal, and S. Baqlakrishnan, A Mobile Transaction Model that Captures Both the Data and the Movement Behavior, *ACM/Balter Journal on Special Topics in Mobile Networks and Applications*, Vol. 2, No. 2, 1997, pp. 149–162.
13. E. Pitoura and B. Bhargava, Revising Transaction Concepts for Mobile Computing. In *Proceedings 1st IEEE Workshop in Mobile Computing Systems and Applications*, 8–9 Dec. 1994, pp. 164–168.
14. Kam-yiu, Lam, Tei-Wei Kuo, Wai-Hung Tsamg, and Gary C. K. Law, Concurrency Control in Mobile Distributed Real-Time Database Systems, *Information Systems*, Vol. 25, Issue 4, June 2000, pp. 261–286.
15. P. Krishna, N. H. Vaidya, and D. K. Pradhan, Recovery in Distributed Mobile Environment. In *Proceedings of the IEEE Workshop on Advances in Parallel and Distributed Systems*, October, 1993, pp. 83–88.