Chapter 3
Localization of Data Sets in Distributed Database Systems Using Slope-Based Vertical Fragmentation

Ashish Ranjan Mishra
Rajkiya Engineering College, Sonbhadra, India

Neelendra Badal
Kamla Nehru Institute of Technology, India

ABSTRACT

This chapter explains an algorithm that can perform vertical partitioning of database tables dynamically on distributed database systems. After vertical partitioning, a new algorithm is developed to allocate that fragments to the proper sites. To accomplish this, three major tasks are performed in this chapter. The first task is to develop a partitioning algorithm, which can partition the relation in such a way that it would perform better than most of the existing algorithms. The second task is to allocate the fragments to the appropriate sites where allocating the fragments will incur low communication cost with respect to other sites. The third task is to monitor the change in frequency of queries at different sites as well as same site. If the change in frequency of queries at different sites as well as the same site exceeds the threshold, the re-partitioning and re-allocation are performed.

INTRODUCTION

Today, the world is witnessing rapid growth in all domains of science and technology, health, agricultural, transportation, manufacturing, commerce etc. All these domains are frequently using the database technology due to the ease of storage of the data. Earlier, when database technology was in its initial stage, the structure of data was very simple and size was very small. So, it is possible only to store the data at the local site. The data can be retrieve and update locally only. Processing of simple data at lo-

DOI: 10.4018/978-1-7998-2491-6.ch003

cal site leads to faster Query Response Time (QRT). QRT of the database is dependent on how fast the data is retrieved from the database repository. As the time passed, the structure of data is evolved to be more complex and size of data has become larger. The user needs the big size of complex data at a higher velocity rate.

The data is often stored in the secondary storage, such as disks. When the database size is smaller, the time spent on reading/writing data from/to disk and operating on the data is usually small. Thus, impact of small sized database on QRT may not be very critical. But, the today database size is getting bigger and bigger. If the database is not organized properly, this will lead to the unacceptable QRT because it takes too much time to search the data in large size databases.

Further the development of distributed system, where group of related people at multiple sites work together to achieve a task, leaded to development of distributed database systems (DDBS). In DDBS, instead of storing the data at single site, data is either fragmented or replicated and stored at various sites. A proper fragmentation and allocation of partitions to proper site can increase the QRT in distributed environment.

There are two partitioning approaches for a relation. First one is horizontal partitioning and second is vertical partitioning. Horizontal partitioning partitions the relation in the smaller relations on the basis of rows. Each smaller relation contains the same number of columns but fewer rows. Horizontal partitioning involves putting different rows into different relations. A horizontal fragment of a relation contains all the rows which satisfy the condition (predicate) applied on relation. Vertical partitioning is a process of dividing the table on the basis of different columns. Vertical partitioning divides a relation into multiple relations that contains fewer columns. In fact, normalization is process of splitting the columns of the table to reduce the redundancy and ease of readability but vertical partitioning is beyond that concept and partitions columns even when they are normalized.

A query does not require the entire attributes of a relation at the same time. Only few attributes of the relation is needed by the queries. So, the vertical partitioning is more effective in improving the QRT rather than horizontal partitioning. Therefore, an attempt has been made here to develop a new slope based vertical partitioning algorithm.

In distributed database, optimal allocation of fragments to the sites also plays a crucial role in improvement of QRT. Queries in distributed database system access the same fragment from many sites. The fragments should be stored in such a way that maximizes the localization of data in the system. The fragments must be allocated to the sites to minimize the amount of the data transfer during the processing of the queries.

The outline of this chapter is organized into as the sections given ahead. Introduction section deals with statement of the problem. It also describes the issues related to vertical partitioning in distributed database. Section 2 describes the previous work related to the vertical partitioning as well as issue and problem of allocation. The chapter describes the previous work in three stages. These are static database vertical partitioning, dynamic database vertical partitioning and dynamic database vertical partitioning. In this section 3, proposed vertical fragmentation model and allocation model have been described. This section 4 describes the performance evaluation of the developed concept. The comparison is made between proposed model and centralized model. At last Section 5describes the conclusions and scopes for future research works.