IRM PRESS
701 E. Chocolate Avenue, Suite 200, Hershey PA 17033-1240, USA
Tel: 717/533-8845; Fax 717/533-8661; URL-http://www.irm-press.com
ITB10721

Chapter XVI

Transaction Processing in Broadcast Databases

Wai Gen Yee, Illinois Institute of Technology, USA

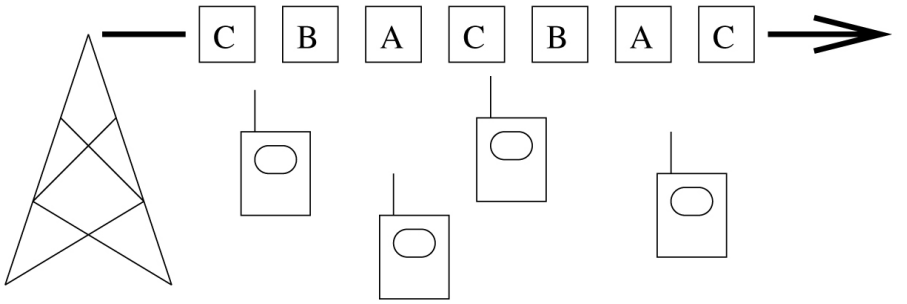
Abstract

Broadcasting is a popular way of disseminating data due to the scalability of its request performance with an increasing population, and its ability to "match" the unique characteristics of modern wireless communications. The necessary decoupling of the clients from the server in the broadcast architecture, however, complicates database-style transaction processing. In this chapter, we describe why broadcast transaction processing is complicated and offer solutions.

Introduction

A broadcast database consists of a single server that cyclically broadcasts a sequence of data items (the *broadcast program*) from the database to the client population. To satisfy a user request, the client scans the *broadcast channel* for the requested item (see Figure 1).

Figure 1. In the broadcast environment, the server cyclically broadcasts the program consisting of items A, B, and C (clients download desired items from the broadcast channel as they arrive)



The main advantage of such databases is request performance that is scalable with the client population. The server achieves scalability because the single broadcast of an item potentially satisfies all outstanding requests for it. Broadcast transmission, however, complicates basic database functionality, such as *transaction processing*. The architecture of broadcast databases precludes conventional transaction-processing techniques. The goal of this chapter is to discuss what complications arise in broadcast transaction processing, and how they can be handled.

Motivation for Broadcast Data Transmission

Conventional database management systems (DBMSs) are unscalable with client populations because they serve each data request individually. For example, during the Clinton impeachment hearings, Web sites such as http://abcnews.com and http://cnn.com experienced severe performance degradation due to heavy request volume (Goodin, 1999). In fact, given a high client-request rate, the degradation of server performance is greater than the proportionate increase in the client-request rate (e.g., doubling the request rate more than doubles the time it takes the server to respond to a request). The single-server queuing theory predicts this phenomenon (Ross, 1997). In response to such phenomena, researchers have designed Web servers that push content to clients (Almeroth, Ammar, & Fei, 1998). These Web servers apply the same general principal as do broadcast databases: increase scalability by simultaneously satisfying multiple data requests.

Broadcast databases are also increasing in relevance due to technological trends. Because client populations are exploding with the spread of mobile and pervasive computing devices (e.g., cell phones, personal digital assistants [PDAs], and laptops), scalability of data services is of prime concern. Furthermore, in order to communicate, these devices are equipped with radio transceivers. Because battery life is at a premium in mobile devices, these transceivers are typically low-powered and more suited to