Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Technology 6 (2012) 640 – 649

Procedia Technology

2nd International Conference on Communication, Computing & Security [ICCCS-2012]

An Efficient and Tolerant Model for Real-time Transaction Management in a Cellular Environment with Random Multiple Handoffs and Frequent Short Disconnections

Abhishek Kayal and Hemanta Kumar Pati*

Department of Computer Science and Engineering, IIIT Bhubaneswar
Bhubaneswar - 751003, Odisha, INDIA

Abstract

In this paper, we present an efficient model for real-time transaction management in a cellular environment with random frequent handoffs and disconnections. In this environment, the Mobile Support Subsystems (MSS) contain special Agents with caching techniques to contain the frequently accessed data from the Fixed Hosts (FH), which can be concurrently or exclusively accessed by transactions being executed in Mobile Hosts (MH) under the radio range of a particular MSS. Each Agent consists of a Cache Coordinator for communication with the Fixed Hosts and the Mobile Hosts and for disconnection and handoff management. The Fixed Hosts consist of a wired network with two types of server – Data Servers and Transaction Servers. The Transaction Servers contain a Transaction Manager for concurrency control and real-time transaction management. Using this schema, the transactions can continue in the Mobile Hosts regardless of random handoffs and frequent short disconnections enhancing real-time transactions and minimizing rollbacks.

Keywords: Mobile transaction; Cache Coordinator; Transaction Manager; Transaction Server; Data Server.

1. Introduction

Terminals, desktop computers, servers are the Fixed Hosts (FH) that are interconnected by means of a fixed network. Large databases run on servers that guarantee efficient processing and reliable storage of data.

* Corresponding author. Tel.: (+91) 9853172288
E-mail address: hemanta@iiit-bh.ac.in

doi:10.1016/j.protcy.2012.10.077

Mobile Hosts (MH) like palmtops, laptops, notebooks, or cellular phones are, of course, not always connected to the fixed network. They may be disconnected for different reasons. The transaction processing models should incorporate the shortcomings of mobile computing such as unreliable communications, frequent disconnections, limited battery power, accidental probability, low bandwidth communications and reduced storage capacity. Frequent aborts due to disconnection should be minimized in mobile transactions. Correctness of transactions executed on both fixed and mobile hosts must be ensured by the operations on shared data. Blocking of mobile transactions due to long disconnection periods should be minimized to reduce communication cost and to increase concurrency. After disconnection, mobile host should be able to process transactions and commit locally. Mobile computing provides the possibility of concurrent access of data by mobile hosts which may result in data inconsistency [3]. Concurrency control methods have been used to control concurrency [12, 17]. Due to limitations and restrictions of wireless communication channels, it is difficult to ensure consistency of data. In this paper, we present an Agent based framework for real-time transaction management and handling of random multiple handoffs and frequent short disconnections with minimum rollbacks while preserving the ACID properties.

The remaining part of this paper is organized as follows. Section 2 summarizes the related research. Section 3 presents the proposed transaction framework for cellular environment. Finally Section 4 concludes this paper.

2. Related Work

In [3-4] transaction models e.g. Reporting and Co-Transactions, Isolation-Only Transactions, Multi Database System Transaction Processing Manager, Weak-Strict Transactions, Two-Tier Replication, Kangaroo Transactions, Pro-Motion, Toggle Transactions, Moflex Transactions, etc are discussed. Most of these models assume long disconnections and working offline, and they relax or redefine the strict ACID properties of traditional transaction model in order to fit to mobile computing environment. Most of them consider mainly totally independent transactions and don't consider the case when the transactions are dependent. In multi version transaction model [6], data is made available as soon as a transaction commits at a mobile host and another transaction can share this data. But data may be locked for a longer time at a mobile host before the lock is released at the database server. In [2], a method based on PLP (Predicted Life Period), which takes care of the dynamicity of the life time of data was proposed. Here, life span of data is predicted based on the probability of data item update. This method makes PLP of data item very close to the actual valid life span of a data item. The above mentioned research works have made little or no attempts on the random handoffs and disconnection issues. In [5], the proposed M-shadow technique, transaction's validation is not tightly coupled to the eventuality of encountering modifications (done by other transactions) on the values of one or more of its data items. Transaction behaviour at run time depends on some characteristics of its set of data items. A new notion called actionability was proposed to describe how a transaction behaves if a value-change is occurred on one or more of its attributes during its processing time and by other transactions. Other than Key attributes (K), actionability classifies the data items used by a transaction into three types: change-accept, change-aware, and change-reject. Here, either, the validation and write phase of the M-Shadow technique is implemented as stored procedures at the primary server, or, the validation and write phase of the M-Shadow technique is implemented as a part of the DBMS, it requires modification in the current existing databases structure. In [1, 18], an Agent-based disconnected transaction model has been proposed. It is based on PLP (Predicted Life Period). The concept of PLP is first introduced in [2]. The model discussed in [1, 18] takes care of the dynamicity of the life time of data. It uses fixed Agents in fixed wired network. A MH connects to servers via these Agents. Frequently accessed data are cached in the fixed Agent situated in the fixed wired network. Whenever an MH enters into a fixed Agent area it can connect and access the data in the cache. But upon update request by a MH, update is done at the local cache and invalidation report is sent to all the mobile hosts which have already accessed the same data. This will force the mobile