Tree-Based Dynamic Primary Copy Algorithms for Replicated Databases

Pradhan Bagur Umesh, A.R. Bharath Kumar, and V.S. Ananthanarayana

Department of Information Technology
National Institute of Technology Karnataka, Surathkal, Mangalore, India
pradhan@ieee.org, a.r.bharathkumar@gmail.com, anvs@nitk.ac.in

Abstract. With increasing demand for performance, availability and fault tolerance in databases, data replication is gaining more and more importance. Resolving or serializing conflicting update requests is the main challenge in large scale deployment of replicated databases. In this paper, we propose the first token based dynamic primary copy algorithms for resolving conflicting requests among different sites of replicated databases. The contribution being reduction in the number of messages required per update request.

1 Introduction

Data replication refers to storage of same data at multiple devices or sites. Availability, high throughput and fault tolerance are the driving factors for replication. Maintaining consistency and resolving conflicting requests are the crucial needs of replicated databases. Conflicting requests arises when there exists multiple requests on a single data item and at least one of them is write or update request. Replicated databases may receive a large number of conflicting requests and hence it is necessary to handle these efficiently such that maximum of one site is updating the data at any point in time and, that site should have obtained the latest data before it starts updating.

Replication strategies can be classified as Lazy and Eager replication [1], [2], [3], [4]. Traditionally, for updating, these use primary copy approach, where any update request is directed to the site holding the primary copy and updates are later propagated to other sites. In dynamic primary copy approach [5], [6], [7] notion of primary copy is dynamic in nature where the update is done at the same site where the request is submitted.

The dynamic primary copy approach in [5] uses lesser number of messages per update with increase in number of conflicting requests. Further performance improvement was proposed in [6] and [7]. In [6] the number of messages per update is of order $\sqrt{N}$. Here, we propose the *first token based dynamic primary copy algorithms* which would be referred as tree-based Lazy Dynamic Primary Copy Algorithm (t-LDPC) and tree-based Eager Dynamic Primary Copy Algorithm (t-EDPC). The t-LDPC algorithm uses number of messages of order log(N) and t-EDPC algorithm uses ‘N’ messages more than t-LDPC but converts the replication strategy Eager. The proposed algorithm has the flavor of Kerry Raymond’s mutual exclusion algorithm [8], but the context of

V. Garg, R. Wattenhofer, and K. Kothapalli (Eds.): ICDCN 2009, LNCS 5408, pp. 362–367, 2009.

the algorithms is different. In [8], the aim was to address the problem of distributed mutual exclusion, whereas here we address the problem of resolving conflicting requests in replicated databases.

The rest of the paper is organized as follows: In section 2, we briefly describe the existing conflict resolution algorithms. We then propose our algorithms in section 3 including the system model and definitions. Analysis of the proposed algorithms is presented in section 4 and we compare our algorithm with the existing algorithms in section 5. Finally we conclude in section 6.

2 Replication Strategies

A distributed database system (DDBS) is a system consisting of data items stored in a set of sites S_1, S_2, S_3,…, S_N and communicate by message passing via underlying communication network. Replicated data in such systems (e.g. Replicated Databases) poses the challenge of maintaining consistency of all the data items and resolution of conflicting update requests. Here, we outline the existing replication strategies

In Primary copy method, data is updated only at the site which hosts the primary copy for that data. The replicas just apply the changes propagated by the primary copy site and all coordination and ordering happens only at primary site. This method has both lazy and eager implementations. CDDR [10] uses this method to update the replicas of data.

Dynamic Primary copy with Piggy-Backing (DPCP) method uses a blend of Lazy and Eager replication. The idea of primary copy is dynamic in nature. The database is divided into pages. When a site wants to update, an update request for a particular page is broadcasted. Upon receipt of permission from all the sites, it updates the page locally.

In Lazy Dynamic Primary Copy (LDCP) and Eager Dynamic Primary Copy (EDPC) methods, the sites are grouped and the communication for each site is restricted to the nodes of its group. EDPC broadcasts each update to all the nodes in the network. LDPC uses Lazy strategy where the updates are multicast to only the nodes of same group.

3 Proposed Algorithm

3.1 System Model

The system has 'N' sites (S_1, S_2, S_3, …, S_N). The underlying communication channel is assumed to be error free and reliable, and message passing between nodes to be asynchronous. Without loss of generality, database is assumed to be fully replicated at all the sites and that each site executes a maximum of one database accessing process.

We assume a non-rooted tree structure of arrangement of nodes in the system which means there exists a unique path any two nodes. Each node can have the knowledge about its neighbors only.