Pseudo-Dynamic Resource Allocation in distributed database

Pseudo-Dynamic Resource Allocation in Distributed Databases

G. Le Lann
Institut de Recherche d'Informatique et d'Automatique (IRIA) Projet SIRIUS,
Rocquencourt 78150 Le Chesnay, France

ABSTRACT

Specific problems arise in distributed processing systems where processors share a common database. Existence of simultaneous updates and retrievals on the database may threaten the database integrity and the consistency of the computations performed on that database.
In this paper, we present a solution to the problem of maintaining the internal integrity of a distributed database. This problem is shown to be identical to a resource allocation problem in a distributed processing system which is subject to failures. A resilient control structure allowing for mutual exclusion is used as an underlying mechanism. A protocol built upon this mechanism is desbribed in detail.

1. INTRODUCTION

In many computing systems, it can be observed that a certain amount of data is shared by processing tasks. It can be just a few variables or a very large data structure. Usually, when the amount of data is large and the semantic rules governing the usage of data are of some complexity, these data structures are referred to as databases.

In this paper, we consider computer networks which include several processors sharing a database. We do not make any assumption regarding the individual computing power of these processors as we do not make any assumption regarding the complexity of the semantics associated with the database.

For our purposes, we assume the database to be dispersed over several storage devices. Also, specific processors are responsible for the physical data handling on these devices ; such processors will be referred to as storage processors.

External users access the computer network simultaneously from different locations. Users can initiate their own tasks and they can also invoke system tasks. We assume that tasks can be partitioned into processing units called processes. Processes are such that, if executed until completion, they meet the database integrity criterion. Specification of the resources they need to get initiated is expressed in terms of data subsets and predicates.

The database is thought of as a collection of items which are lockable individually. Examples are file subsets or individual records. We do not make any assumption regarding the size of these items. Choice of a particular size is generally dictated by the nature of the applications using the database and by the degree of acceptable bookkeeping overhead.

User and system processes are run on several processors which perform the analysis of process needs. On behalf of processes, they are allowed to read data items, even those which are locked.

From the analysis of predicates on data subsets as provided by a process and from the reading of data items, a processor can infer which data items are actually necessary for a process to be initiated. Let us point out that there is no assumption made so far regarding the strategy used by processes to request data items. Processes may state their needs a priori or they may request new data while running, i.e. dynamically.

These processors are also provided with an internal mapping mechanism allowing for the identification of which storage processor is to be accessed to reach a given data item. Such processors will be referred to as database controllers or controllers.

As far as users are concerned, the existence of several controllers, storage processors and devices should be kept invisible. Actually, the whole computer network is to be viewed as a single system.

Communications are message-based. We will assume that interprocess transit delays are variable and that transmission errors are possible. All communications are handled by a specific piece of software. Examples for packet-switching networks are the ARPA NCP (1) and the CYCLADES TS (2).

Following this introduction, a description of two different system architectures is given along with a presentation of some