PERFORMANCE EVALUATION ON SEVERAL CAUTIOUS SCHEDULERS
FOR DATABASE CONCURRENCY CONTROL.

S. NISHIO*, M. WATANABE**, Y. OHIWA***, T. HASEGAWA

Department of Applied Mathematics and Physics, Faculty of Engineering
Kyoto University, Kyoto, 606 Japan

ABSTRACT: In recent studies on concurrency control of database systems, "cautious schedulers" have been developed intensively because they avoid both abortion and rollback in the process of concurrency control. In this paper, by means of simulation studies, we will evaluate the performance of three cautious schedulers: (1) cautious two-phase locking scheduling, (2) exclusive preclaimed two-phase locking scheduling, and (3) cautious conflict serializable scheduling. The performance of these cautious schedulers is compared with that of corresponding non-cautious schedulers.

Key Words: Concurrency control, Cautious scheduler, Performance evaluation

1. INTRODUCTION

In database systems a commonly used criterion of correctness of concurrent execution of transactions is serializability [3, 9], and many different schedulers for concurrency control have been proposed to achieve serializable executions of transactions [2, 19]. In this paper the performance of several concurrency control schedulers is evaluated via simulation experiments aimed at comparing the perform-

* Dr. Nishio's former publications are under the name Shojiro MURO.
** Mr. Watanabe is now with the Systems Development Center, Canon Inc., 30-2, Shimomaruko 3-Chome, Ohta-ku, Tokyo, 146 Japan.
***Miss Ohiwa is now with the Central Research Laboratory, Matsushita Electric Industrial Co., Ltd., 3-15, Yagumo-Nakamachi, Moriguchi, Osaka, 570 Japan.

157

M. Kitsuregawa et al. (eds.), Database Machines and Knowledge Base Machines
ance of so called **cautious concurrency control schedulers** (e.g., [4, 7, 8, 12, 13]; for a survey study see [14]) with that of usual (non-cautious) concurrency control schedulers.

In cautious scheduling, it is assumed that the information about the future operations of a transaction is provided to the scheduler when the first operation of the transaction arrives at the scheduler. Taking this information into account, the scheduler examines the possibility of non-serializable execution for each input operation. If any such possibility exists, the scheduler delays the execution of the operation and executes the next operation. In such a way, once the scheduler accepts the first operation of a transaction, the operations of the accepted transaction will not be **rolled back** or **aborted**, but will be ordered as a serializable sequence. For this reason the cautious scheduler is attractive for database users. However, the conservative execution of operations reduces the degree of their concurrent execution. On the other hand, in the usual concurrency control schedulers, rollback or abortion would occur due to **deadlock** or non-serializable execution of operations. A recent study by Carey [5] indicates that frequent rollbacks considerably reduce system performance. Thus, a comparative study of the degradation of concurrency in cautious schedulers due to conservative execution and the degradation of throughput in non-cautious schedulers due to rollback or abortion will be of interest to the database system designer.

In this paper we consider the following three cautious schedulers: (1) **cautious two-phase locking**, (2) **exclusive proclaimed two-phase locking** [5], and (3) **cautious conflict serializable scheduling** [7, 8]. For performance comparison, two non-cautious schedulers corresponding to above three cautious schedulers, i.e., (4) **two-phase locking** [9] and (5) **conflict serializable scheduling** [1, 11, 17] are also considered.

On the basis of several performance criteria such as throughput and waiting time in several queues, we will evaluate the performance of these five schedulers, and discuss the characteristics of each. In particular, we will discuss the advantage of cautious schedules through the comparison of cautious and non-cautious schedulers.