Chapter 7

The Table Oriented Approach

This chapter presents recovery strategies relating to the table layer (see section A.4), using tuple logging combined with concurrency control supporting fine granularity and semantically rich tuple locking (see section A.9). The table oriented recovery approach satisfies a larger set of recovery design goals (see section 2.6) than the compensation oriented approach, which is the other of the previously presented recovery approaches that supports fine grained and semantically rich locking (see chapter 6).

The table oriented recovery approach is not fully used in any commercial DBMS, to our knowledge. Policies from this approach are on the other hand used in some modern DBMSs such as Encompass ([Bor81], [Hel85a], [Hel85b]), Tandem Non-Stop SQL ([BP88], [Gro88]), and TechRa ([HL86]).

7.1 The Table Oriented DBMS Server Architecture

This section presents the table oriented DBMS server architecture (see figure 7.1). The presented architecture does not correspond to any particular implemented DBMS and is used to describe and analyse recovery capabilities of the table oriented approach. The main difference between the table oriented DBMS server architecture and the compensation oriented DBMS server architecture (see section 6.1) is that concurrency control and recovery services are performed at the table layer instead of to the record and block layers. In addition, a separate access method server is introduced at the table layer.

The concurrency control server receives transactions (see section A.8) and tuple DB operations (see section A.5) from the relational server. The concurrency control server serialises the tuple DB operations by using fine grained, semantically rich tuple locks according to the specifications given in section A.9. It is assumed that for every table a primary key or (for those relations where a primary key is not defined) a surrogate key is defined (see section A.4). The term primary key is, in this chapter, also used for a surrogate key. The primary key specification is part of the dictionary representation of the table.

A write, insert, or delete tuple operation write locks the involved tuple on its primary key value. A delta operation delta locks the involved tuple on its primary key value. A read operation read locks the involved tuple on its primary key value. The locks are well-formed strict two-phased (see section A.9).

The access method server maps a tuple to its record representation. A tuple is assumed to be mapped to one record and is therefore contained within one block (see section 6.3.1). The access method scheduler supports block granularity traditional locking, i.e. access method block (AM) locks. An AM read lock is kept over the duration of a tuple operation to a block, i.e. the duration.
Figure 7.1: The table oriented DBMS server architecture. The layered DBMS architecture corresponds to the architecture presented in section A.3 and figure A.3. The concurrency control server is included in the table layer. The access method server included in the table layer maps a tuple to its record representation. The access method scheduler serialises access method operations.