ABS Tech Note: DAL18 Rdb Locking Issues (6/92)

Article Created: 30 June 1992

TOPIC --

This technical note discusses issues involving DAL performance that are
specific to the Rdb database on VAX/VMS.

DISCUSSION ---

DAL provides a model for specifying the type of "protection" or "locking"
desired when using a database or specific table. Both the OPEN DATABASE
and OPEN TABLE statements allow for update modes (READONLY/UPDATE) and
sharing modes (SHARED/PROTECTED/EXCLUSIVE). The default mode is UPDATE and
PROTECTED.   Typically this would mean that the user has a "READ/WRITE"
lock on the particular table or database. The sharing and update modes are
advisory and are passed on to the DBMS only WHEN APPROPRIATE.

Rdb provides a some what different model for specifying protections that
does not map well with the DAL model. Rdb uses an extension to SQL
transactions — the SET TRANSACTION statement — to signal the type of
locking mechanism desired for that transaction. SET TRANSACTION modes are
not persistent across subsequent transactions; they must be specified at
the beginning of each new transaction or the default mode of "READ WRITE"
and "SHARED" will be used.

DAL update and shared modes are not transaction based; that is, in the DAL
model, update and shared modes are "global" session modes and are
independent of beginning and committing sessions. Consequently, when a DAL
user opens a Rdb database or table, specifying a particular update or
shared mode, the Rdb transaction modes are not changed. The DAL Server
does not alter the Rdb transaction modes because they can not be changed in
the middle of a Rdb transaction. One would have to commit the current
transaction and begin a new transaction, setting the transaction modes as
desired.

There are two possible problems:

Problem: Lock Conflicts caused by inserts and updates while using the Rdb
          default transaction modes.
Solution: Commit as early and often as possible. Generic DAL applications

```
             should commit often to free any locks against tables and indexes
             accessed. Otherwise, other users might be locked out of one or
             more tables.

Problem:  User wants to specify READ ONLY to prevent locking.
Solution: Applications tailored to Rdb can explicitly begin new
          transactions with the desired locking and update modes by using
          the Rdb SET TRANSACTION statement via the DAL EXECUTE IMMEDIATE
          statement. Below are examples:

          open rdb dbms;
          open rdb database "daldemo";
          execute immediate "SET TRANSACTION READ ONLY;" in Rdb;
          ...queries....
          commit;                  /* ends this transaction with READ ONLY */
          insert into ....
          commit;
Copyright 1993, Apple Computer, Inc.


Keywords:  <None>

===================================================================

This information is from the Apple Technical Information Library.

19960215 11:05:19.00
```

Tech Info Library Article Number: 11645