
IMAGE/SQL: Part 3 of N

Fred White

Adager Corporation

Sun Valley, Idaho 83353-3000 • USA

<http://www.adager.com>

In Part 2 of N, I discussed global defects in the program IMAGESQL.PUB.SYS (IMAGESQL) which make it difficult for Database Administrators (DBAs) to maintain their SQL DBEnvironments (DBEs) and their IMAGE/SQL databases (DBs).

Rather than advocating a total redesign of the DBE data structures and the IMAGESQL command set, I resigned myself to accepting the current design and simply recommended modifications intended to minimize the impact of these global defects and make the DBA's job easier.

In this part I discuss the current (July 1994) versions of the IMAGESQL commands “SET SQLDBE”, “SET TURBODB”, “ATTACH” and “ADD USER”.

Acceptance

“SET SQLDBE” is a parameter-setting command to which subsequent DBE-relative commands are applied. Implementation flaws include:

1. If there is no DBE with the specified **DBename**, IMAGESQL offers to create a DBE fileset with that name even though it will be unable to do so due to a file naming conflict.

IMAGESQL should simply connect to the named SQL DBE. If none exists, IMAGESQL should report that to the user *without* logging the command.

The command should have an optional “;NEW” parameter for creating a new DBE. If

SET SQLDBE

the “;NEW” parameter is present, IMAGESQL should verify that there will be no file naming conflict *before* attempting to create the DBE fileset. Currently, it “blindly” proceeds and, if a naming conflict exists, quits *without* purging any of the (privileged) files it created. If the creation fails (for whatever reason), IMAGESQL should purge whatever files have already been created before returning to the user.

No log record should be written unless the command is successful. The “;NEW” parameter, if present, should be excluded from the SET SQLDBE log record so that, when the log file is subsequently used as an XEQ file, the logged SET SQLDB command will be treated as a reference to an *existing* DBE rather than as a request to create a *new* DBE.

2. If you are not the DBE creator (DBEC) and if the DBEC has not granted DBA authority to you, SET SQLDBE fails.

This is OK except for the fact that IMAGESQL provides no command for the DBEC to grant you DBA authority. Instead, the DBEC is compelled to use ISQL (with its own command set, syntax, mediocre help facility and set of manuals) to grant you DBA authority.

SET TURBODB

“SET TURBODB” is a parameter-setting command which (as recommended in Part 2 of N) should be available only to the database creator (DBC).

IMAGE is a top-down system. It is important for the DBC to review and perform *all* IMAGE-to-SQL mappings at the database (DB) level. (Refer to Part 2 of N where I suggested a “MAP” command for providing the functionality of the current UPDATE TYPE and SPLIT commands at the DB level.)

The ideal place for keeping DB level mapping information is in the root file. The SET TURBODB command should *automatically* generate that table (if not already present) and all *manual* mappings should be performed prior to attachment to a DBE.

(Note that the SET TURBODB command *fails* if the root file is currently being accessed exclusively or if the database is being accessed with a DBOPEN mode of 2, 3, 4, 6, 7 or 8.)

“ATTACH” is another command that only the DBC should be allowed to use (as mentioned in Part 2 of N). It has several defects:

1. Even though the current DB and the current DBE are in different accounts, IMAGESQL attaches the DB to the DBE.

Attaching across accounts has disastrous implications for DBAs relative to keeping the files/databases of a DBE in synch. Either the designers were unaware of the consequences or were unwilling to impose constraints on users or simply neglected to do so. At the very least, IMAGESQL should warn you of the potential problems. Actually, with a well designed system (which I will present in future columns) there is no need for such a capability.

Please avoid using this “feature”.

2. IMAGESQL allows you to attach your DB to more than one DBE.

This too has disastrous implications for keeping the files/databases of a DBE in synch and is yet another negative consequence of poor design. It led IMAGESQL to keep the names of the DBEs to which a DB is attached in a privileged file named <DBname>TC rather than keeping a single DBE name in the DB's root file.

This TC-file is just another file to maintain and get out-of-synch. For example, DBSTORE/DBRESTOR both support this TC-file but DBUTIL suffers from the side effects of its existence. It purges an attached DB without warning you that the DB is attached and without purging the (privileged) TC-file.

Please avoid using this “feature”.

3. A set of default “views” (one for each dataset of the DB) is ***automatically*** created for the DBC based on the usually false assumption that the DBC will want to perform “creator” SQL access to ***all*** of the datasets of the DB.

In the real (commercial data processing) world, DBCs are interested in providing SQL access for others. If they should want it for themselves, they need only perform one ADD USER command.

In any event, this “view creation” is premature since the views of all datasets referenced by

subsequent UPDATE TYPE or SPLIT command will require modification.

4. The ATTACH command fails if the DBE is “in use” (i.e., an SQL process is “connected” to the DBE).

Since a DBE can have several DBs attached to it, this means that the DBE must be “stopped” (using ISQL) whenever you wish to attach another DB to the same environment. This stoppage can be annoying to SQL users who were accessing the DBs already attached to the DBE.

ADD USER

“ADD USER”, in addition to needlessly requiring a DBOPEN PASSWORD (see Part 2 of N), has the following defects:

1. The DBC, in addition to entering a user name, must enter the command string “ADD USER” along with the access class (PASSWORD) and MODE for every user being added.

If a new parameter-setting command such as “SET USERGROUP” (a la SET TURBODB) were provided, the DBC could be iteratively prompted with “USER NAME >>” and respond with only a user name. When entering hundreds of names, this saves numerous keystrokes and typing errors.

The DBOPEN mode and DBOPEN access class would be parameters of the new “SET USERGROUP” command which would provide an optional “;NEW” parameter to be treated in a manner entirely similar to the manner it is treated as a parameter in the SET SQLDBE command described earlier.

The “views” pertinent to an SQL usergroup are created when the USERGROUP is created and are derived from the DB level mapping table created by the SET TURBODB command as modified by subsequent MAP commands.

2. The ADD USER command *fails* if any process is “connected” to the DBE.

This means that all SQL users must terminate their connections to the DBE and remain disconnected while the DBC is adding new (possibly hundreds of) SQL users. This defect would be more readily avoidable if SQL usergroups could be created in a DB-independent (and

DBE-independent?) manner. (See defect 3, which follows).

3. There is no provision for “wild card” characters and MPE group names within the SQL username.

Most production environments have their users log on (sometimes automatically) into just a few MPE groups. If the SQL username syntax supported MPE group names *and* wild cards, adding tens or, in some cases, hundreds of users could be accomplished with a single ADD USER command. (This technique was implemented in the mid-70s for IMAGE remote database access.)

In addition to minimizing the SQL user table sizes, SQL users can be “added” or “deleted” *externally* to IMAGESQL simply by using existing MPE commands to delete/add users and/or to modify home groups and/or to modify logon facilities. This method of maintaining SQL user tables can be performed without requiring the DBE to be “stopped”.

The most used wild card character in the HP3000 world is the commercial at sign (@). The use of this character in any other manner within IMAGESQL commands should be eliminated.

4. Currently, user names *require* inclusion of an account name. This is an accidental by-product of the fact that DBs are allowed to be attached across account boundaries which, itself, is an unnecessary and undesirable “feature” as mentioned earlier.

Even if all of your SQL users are within your own logon account, you must enter your account name in each ADD USER command. If you accidentally enter some other valid account name, you have erroneously authorized some unknown user SQL access to your DB.

5. Whenever an ADD USER command results in the creation of a new SQL usergroup, IMAGESQL creates views for each dataset of the DB to which that group of users has access as determined by the DBOPEN mode and DBOPEN access class (PASSWORD). This is NOT a flaw but DBCs should be abundantly aware of this “feature”.

To eliminate the creation of unnecessary views and restrict SQL access to just the desired datasets, the DBC should add new access classes

to the DB and modify the read class and write class lists of the selected datasets before attaching the DB to a DBE. The DBC can then reference only these newly added access classes (PASSWORDs) within his ADD USER commands.

To be continued ... To be continued ...