

Considerations for a SAS 6.06 AF/FSP and DB2 Application
Kent Andrew Spencer
Halliburton Company Information Services Center

ABSTRACT

The new version of SAS gives developers powerful links to DB2. Now, manipulating information in third party data bases is nearly as easy as manipulating information in SAS data bases. SAS PROC's for producing graphs and statistics can now access dynamic DB2 tables. Unfortunately, we are not dealing with true SAS data bases, and certain limitations exist. This paper discusses the issues address in developing Halliburton Company's Employees' Trust Contribution Reporting System.

HISTORY

The Employees' Trust Contribution Reporting System is designed to provide users at remote locations the ability to update instantly the corporate offices' records of contributions to the employees' trust. This eliminates the re-keying of information from forms sent to the corporate office. The system would have a menu driven interface with context sensitive help. The application would have around 20 concurrent multiple users updating at various sites across the nation.

Nothing in the specifications for the application requires the use of DB2 as the storage medium for the data. DB2 was chosen to hold the information because SAS/Share for version 6.06 was not scheduled for release.

ISSUE ONE

Are BASE SAS, SAS/AF, and SAS/FSP running on the same machine as DB2? If these products are not licensed for the same CPU, the licensing agreement must be changed and software migrated so these products are on the same machine.

ISSUE TWO

Referential integrity must be off because SAS/FSP attempts to update the primary key, and you must have a primary key to implement DB2's referential integrity. An attempt to update the primary key returns a DB2 error code of -535. The same problem is encountered when SAS/FSP tries to update the key of a partition, returning a -151 error. The key field cannot be a protected, hidden value on the screen since SAS/FSP will still try to update it. Also, the column cannot be left off the screen - SAS/FSP requires that it be on the screen if the row is acted on.

Considerations for a SAS 6.06 AF/FSP and DB2 Application
Kent Andrew Spencer
Halliburton Company Information Services Center

ISSUE THREE

Page locking may result when SAS/FSP tries to update a DB2 row. This can occur because SAS/FSP holds the corresponding page in the DB2 table. Since SAS does not have a facility to issue a DB2 commit, other users attempting to access adjacent rows are locked out. Resultingly, special design considerations are necessary for DB2 tables that SAS/FSP updates.

1. Each record must have a unique key. This unique key helps organize the data into distinctive groups. Thus, a reduction on the likelihood of page locking occurs if the users accessing the information tend to only access their own group of information.
2. There should be enough rows in the DB2 table to decrease the likelihood of finding rows from different groups of data on the same page. To accomplish this, loading large amounts of historical data in batch before the system goes into production is necessary. For our application, our initial load consisted of 1500 rows of 53 columns.
3. The DB2 table should be laid out such that records with similar keys get inserted next to each other in the DB2 table. This is another measure to prevent page locking with other users.
4. There must be sufficient free space in the table for inserting like groups of records next to each other. Otherwise, the likelihood of page locking increases when inserting rows into the table.
5. There should be an abundant number of columns per row, or the columns must be sufficiently long, so that a very low number of rows occur on a page in the DB2 table. The number of rows that occur on a page can be further reduced by increasing the free space, but this can quickly become inefficient. Too few columns result in many records per page and a greater likelihood of page locking.

This was a pilot project with our goal being the successful interaction of SAS AF/FSP with DB2 using the SAS/Access product. These design considerations do not imply that this is the only possible DB2 table design, that this is a good DB2 table design, or that this is a good DB2 table design for a particular DB2 site.

STATUS

The application has been in production since Jan 1, 1991, with users conducting about 275 inserts to the table per month. People

Considerations for a SAS 6.06 AF/FSP and DB2 Application
Kent Andrew Spencer
Halliburton Company Information Services Center

are very impressed with the look and feel of the SAS FSP/AF front end to the DB2 tables. Although extensive documentation was provided with the application, many user have expressed that the documentation was unnecessary.

People have not had any page locking problems. This greatly surprised me since I had expected some problems with page locking. To date, not one user has called to complain that he could not access a record that he felt certain he had added. I feel this good fortune is principally attributable to the small number of rows inserted into the table each month.

The application also makes extensive use of SCL to generate reports on-line based on user defined criteria. This ability allows the users to have the reassuring feeling of "paper in hand."

CONCLUSION

SAS 6.06 is an excellent tool producing graphs and reports directly from a DB2 table, using SAS/Access as the interface. Still, there are some serious issues to consider when a many users are updating data in the table, or when there is a small amount of data in the table. The development of this application also proved to us that the Screen Control Language is a robust tool for giving a mainframe application a PC look and feel. Finally, PROC SQL is an excellent addition to the library of SAS procedures.