



# Craig S. Mullins & Associates, Inc.

*Database Performance Management*

[Return to Home Page](#)

March 1996



## DB2 Version 4 Security and Authorization Enhancements

*By Craig S. Mullins*

Although DB2 Version 4 provides a wealth of new functionality for administering DB2 databases, performance and availability enhancements seem to be the only new features being discussed in any depth. But IBM has made several useful security and authorization enhancements in DB2 V4 that will prove helpful to users of DB2.

### **Authorization for dynamic SQL applications**

Traditionally, authorizing individuals to use dynamic SQL against DB2 tables has posed one of the biggest security dilemmas for DB2 administrators. Prior to DB2 V4, any user of dynamic SQL, whether

embedded into an application program or issued dynamically via SPUFI, QMF, or another query tool, had to have explicit authorization to issue the SQL statement in question. Additionally, the user must have been granted the EXECUTE authority for the plan being run. If the user does not have the requisite authority, DB2 will prohibit the access, rollback any database changes, and generate an error message. All dynamic SQL authorization errors are detected at run time. This has the potential of causing an enormous amount of administrative overhead. Consider the situation depicted in [Figure 1](#). A program is coded that uses dynamic SQL to insert, select, update, and delete from several different DB2 tables. The following GRANT statements must be issued for each and every user of the program:

```
GRANT EXECUTE ON plan TO USER;  
GRANT INSERT ON TABLE_A USER;  
GRANT UPDATE ON TABLE_B TO USER;  
GRANT SELECT ON TABLE_C TO USER;  
GRANT DELETE ON TABLE_D TO USER;  
GRANT INSERT ON TABLE_E TO USER;
```

As the number of users increases, the administrative burden of maintaining the authorization needs of the application also increases. But another, more insidious problem lurks within this implementation. The intent of the series of GRANT statements listed above is to enable a user to execute a specific DB2 application program. However, by granting the user table authorities, that user is capable of bypassing the

program altogether and issuing ad hoc, dynamic SQL statements using a query tool like SPUFI or QMF. In many cases, organizations wish to deliver the capability of using dynamic SQL within the scope of a specific program, but prohibit it outside of that protected scope.

DB2 V4 provides relief from this archaic dynamic SQL authorization scheme. A new BIND parameter, DYNAMICRULES, had been added to provide administrators an option when granting security for users of dynamic SQL applications. This parameter enables the programmer or DBA to indicate whether authorization checking should occur at BIND time or at run time.

If the DYNAMICRULES parameter is set to "RUN," DB2 will perform authorization checking as it did in all prior releases — during PLAN execution. The user will still need authorization to access and/or update all tables accessed by dynamic SQL. However, if the DYNAMICRULES parameter is set to "BIND," DB2 will perform authorization checking during the BIND process, not at run time. The authorization ID used to verify security during BIND process will be the value of the current SQLID at BIND time. So, in this new scenario, only the binder requires authorization to access the tables. Users of the plan can exercise all the privileges of the plan owner and therefore only require EXECUTE authority on the plan. This can greatly simplify the administration of programs

containing dynamic SQL.

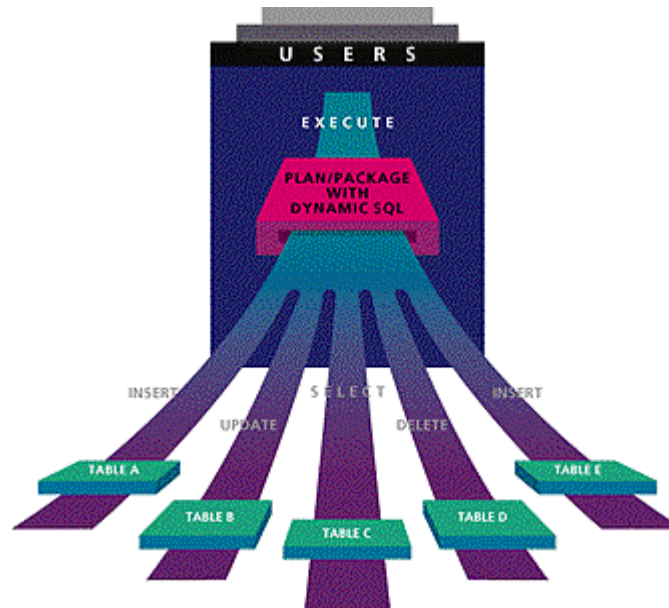


Figure 1 - Dynamic SQL application

### **Modifying the behavior of BINDADD**

DB2 V4 provides several other useful modifications to its security structure. One such enhancement provides a way to alter the meaning of the BINDADD privilege using a new DSNZPARAM (DSN6SPRM BINDNV), otherwise known as "BIND NEW PACKAGE." If the BIND NEW PACKAGE parameter is set to "BINDADD," the BINDADD privilege will function as it did in prior DB2 releases. "BINDADD" is the default.

If the "BIND NEW PACKAGE" parameter is instead set to "BIND," two changes will take effect:

- Users with PACKADM authority will no longer require BINDADD authority to BIND a new package or package version
- Users having the BIND of PACKADM privilege will be able to BIND new package versions

### **Referential security**

So, DB2 V4 has modified the functionality of a privilege, but it also provides a new privilege, namely the REFERENCES privilege. This privilege has been added to enable DBAs to better administer referential integrity. A user granted the REFERENCES privilege can create or drop a referential constraint in which the named table is the parent table.

The diagram in [Figure 2](#) shows a parent/child relationship between the COURSE table and the INSTRUCTOR table. Assume that there is a one-to-many relationship between courses and instructors such that an instructor can teach many courses, but each individual course can be taught by only one instructor. If user "Joe" is granted REFERENCES privilege on the INSTRUCTOR table, then "Joe" can create a foreign key in the COURSE table that references the primary key in the INSTRUCTOR table.

It is important to note that the existence of the REFERENCES privilege on a table does not change the meaning of the ALTER privilege. Holders of the ALTER privilege on a table will still be allowed to create or drop a referential constraint on the named table. The REFERENCES privilege gives the grantee

the ability to create a referential structure without also having the authority to perform other types of database changes.

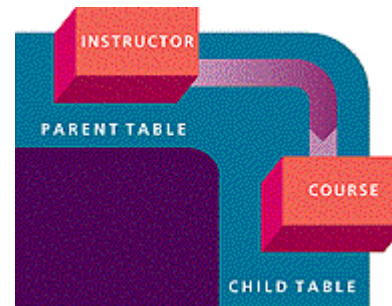


Figure 2 - A sample parent/child referential structure

### **Termination authorization**

In previous DB2 releases, only users with SYSOPR, SYSCTRL, or SYSADM authority could use the TERM UTILITY command to terminate another user's utility job. In DB2 V4, a holder of the DBMAINT (and therefore DBCTRL and DBADM) privilege can use the TERM UTILITY command to terminate a utility job.

The user of the TERM UTILITY must have sufficient authority over every database listed in the job. If not, the command has no effect on any part of the job. It may be necessary to plan separate utility jobs for separate databases in order to make best use of this expanded privilege.

### **Synopsis**

The changes to DB2's authorization structure for V4 can significantly ease the security aspects of DB2

database administration. Wise DBAs and security analysts will learn and implement these features as they migrate to DB2 V4.

*EDGE Magazine Online.*

© 2005 Mullins Consulting, Inc. All rights reserved.  
[Home](#). Phone: 281-494-6153 Fax: 281-491-0637