

Test Individual SQL Statements Fast

Business Challenges

As companies strive to increase revenue and improve customer service it becomes increasingly important to create or modify DB2 applications as quickly as possible.

Although developers may create or modify only a small portion of an application, testing the modification still requires the time consuming process of compiling and executing the entire application.

Test Application SQL While Coding

When developing and coding a DB2 application every developer wants to know if they are coding valid SQL and if the SQL will return the results they expect.

SoftBase's **DB2 SQL Debug** component of TestBase lets a developer code and dynamically test SQL statements by pointing to them while editing or viewing an SQL program. From there they can execute and / or Explain the SQL without having to compile and execute the application. All DB2 developers know the importance of good access paths.

DB2 SQL Debug provides faster, easier, and more reliable SQL statement testing and performance analysis.

DB2 SQL Debug Makes Applications More Reliable

SoftBase's DB2 SQL Debug allows individual SQL statements to be tested while coding the application program – before the program executes. DB2 SQL Debug can help developers quickly assess the quality of existing, new, or altered SQL statements during development. This helps developers catch potentially problematic SQL statements early in development when they are easier to optimize.

Point to the SQL while in Edit or View

Use the QQ line commands to point to the SQL you want to execute. The command can be used in block form as shown or Qn.

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT TST.G610.TDGF.LIB.SOURCE(S3930000) - 01.11 Block command inco
Command ==>
000524 MOVE S3-TDAT TO TDAT-TDAT-NAME Scroll ==>
000525 MOVE S3-ENVIRONMENT-CODE TO TDAT-ENVIRONMENT-CD
000526 MOVE S3-SYSTEM-CODE TO TDAT-SYSTEM-CD
000527 EXEC SQL
QQ SELECT *TIMESTAMP (TDAT_DATE , TDAT_TIME)
000529 INTO SW-CURRENT-TIMESTAMP
000530 FROM TEST_DATA
000531 WHERE ENVIRONMENT_CD = :TDAT-ENVIRONMENT-CD
000532 AND SYSTEM_CD = :TDAT-SYSTEM-CD
000533 AND TDAT_NAME = :TDAT-TDAT-NAME
QQ
000534 END-EXEC
000535 IF SQLCODE NOT = ZEROS
000536 MOVE 'ERROR SELECTING DATE FROM TDAT'
000537 TO S3-ERROR-MESSAGE
000538 MOVE +0002 TO S3-ERROR-NUM
000539 SET S3-ERROR-TYPE-DB2 TO TRUE
000540 PERFORM 990000-ERROR
000541 END-IF
    
```

- Improve Developer Productivity with Faster Debugging
- See Statement Access Paths without Compiling and Executing Application
- Reduce Development and Testing Time

DB2 SQL Debug Program SQL Options

This panel prompts for the subsystem, how the SQL should be gathered from the program and whether the SQL should be executed or explained or both.

```

QBase Program SQL Options
Command ==>
Enter SQL Option Values:
DB2 Subsystem Name . ==> DBCG DB2 Subsystem Name
DB2 SQLID . . . . . ==> S3DBA Plan Table Qualifier
DB2 Schema Name . . ==> TDGF610 Qualifier for Unqualified Tables
Commit or Rollback . ==> R C - Commit R - Rollback
Maximum Result Rows ==> 01000 Maximum Number of Result Rows to Fetch
Explain Query Number ==> 00001 Query Number for Explain in Plan Table
Left Margin . . . . . ==> 00008 First Source Column to Process
Right Margin . . . . . ==> 00072 Last Source Column to Process
Execute . . . . . ==> Y ( Y / N ) Execute SQL
Explain . . . . . ==> N ( Y / N ) Explain SQL
    
```

DB2 SQL Debug Host Variable Prompt

Here you can supply the host variable values before executing the SQL. Data type is shown so there is no guessing or reading of application code to determine. No quotes are needed for character data since it is a fully prepared SQL statement.

```

QBase Program SQL Host Values
Command ==>
DB2 Subsystem Name . ==> DBCG Host Variable 0001 of 0003
DB2 Plan Name . . . . . ==> TDGF610 Scroll ==> PAGE
DB2 Schema Name . . . . . ==> TDGF610 DB2 SQLID . . . . . ==> S3DBA
Press END key to continue. Press CANCEL to cancel.
Enter SQL Variable Values:
Host Variable / Data Length& Nulls
Value Type Scale
-----
000001 TDAT-ENVIRONMENT-CD SMALLINT 0000002 Y
+00201
000002 TDAT-SYSTEM-CD SMALLINT 0000002 Y
+00004
000003 TDAT-TDAT-NAME CHAR 0000008 Y
CSB1
    
```

DB2 SQL Debug Results

Results are shown in Testbase View

```

Testbase Table Processor
Command ==>
width= 80:0018 DB2 Location ==> DBCGLOC DB2 Subsystem ==> DBCG
DB2 Object. . ==> SEE INPUT SQL FOR TABLE(S) current Sqlid ==> S3DBA
Columns ==> 1 Thru 26 of 26 IX ==> Lobs ==> N
-----1-----2-----
000001 2016-07-26-16.25.48.000000
    
```

DB2 SQL Debug

DB2 Access Path In DB2 SQL Debug

This shows the access path chosen for the SQL statement. It is the data from the plan table..

```

QBASE Access Paths                                     Lines 1 of 3
Command ==> |                                         Scroll ==> PAGE
                                                    DB2 Subsystem ==> DBCG

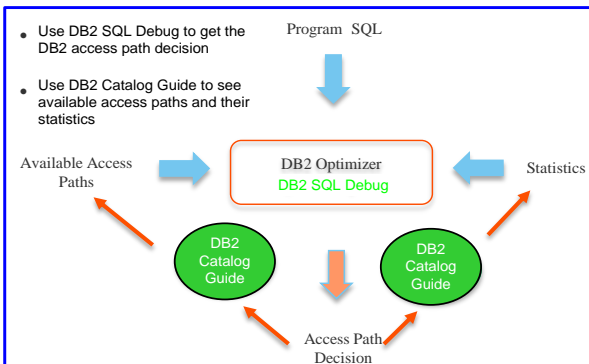
S                                     LM SORT SORT
SE L M S B A                         OO NEW COMP
UL P                                     X - Expand
BE L M S B A                         / - List commands
C A IOE N                             S - Select
T N XPQ O TT AT IX MC M F E KE UJOG UJOG
01 01 000 01 T I N 03 0             IS NNNN NNNN
TABLE: TDAT
INDEX: TDAT01CP
    
```

In this case we have an index access matching on 3 columns.

If also licensed for TestBase DB2 Catalog Guide or TestBase View / Edit, those commands can be issued next to the table or the index. Using DB2 SQL Debug in conjunction with DB2 Catalog Guide allows for viewing all 3 determinants of access path selection:

1. SQL Statement
2. Access Paths Available
3. Statistics on those Access Paths

Use DB2 Catalog Guide with DB2 SQL Debug



DB2 Catalog Guide Index Information

```

TestBase DB2 Index                                     Lines 1 of 1
Command ==> |                                         Scroll ==> PAGE
                                                    DB2 Subsystem ==> DBCG

DB2 Location: DBCGLOC
DB2 Object : TDGF610.TDAT01CP

Commands: Find      Export      Explode

NAME . . . . . TDAT01CP
CREATOR . . . . . TDGF610
TBNAME . . . . . TDAT
TBCREATOR . . . . . TDGF610
UNIQUERULE . . . . . P
COLCOUNT . . . . . +00003
CLUSTERING . . . . . Y
CLUSTERED . . . . . Y
DBID . . . . . +00274
OBID . . . . . +00004
ISOBJID . . . . . +00005
DBNAME . . . . . TDGF6140
INDEXSPACE . . . . . TDAT01CP
FIRSTKEYCARD . . . . . -0000000001
FULLKEYCARD . . . . . -0000000001
NLEAF . . . . . -0000000001
NLEVELS . . . . . -00001
BPOOL . . . . . BPO
PGSIZE . . . . . +00004
ERASERULE . . . . . N
DSBTPASS . . . . . N
CLOSERULE . . . . . N
SPACE . . . . . +0000000000
IBMREQD . . . . . N
CLUSTERRATIO . . . . . +00000
CREATEDBY . . . . . CSBJ
IOFACTOR . . . . . +00000
PREFETCHFACTOR . . . . . -00001
STATIME . . . . . 0001-01-01-00.00.000000
INDEXTYPE . . . . . 2
FIRSTKEYCARDF . . . . . -1.000000000000000E+00
FULLKEYCARDF . . . . . -1.000000000000000E+00
CREATEDTS . . . . . 2016-04-22-16.42.37.215619
    
```

This shows all the data in SYSIBM.SYSINDEXES

DB2 Catalog Guide - Indexes on a Table

Placing an IX command next to the table we would get the following:

```

TestBase DB2 Objects                                     Lines 1 of 2
Command ==> |                                         Scroll ==> PAGE
                                                    DB2 Subsystem ==> DBCG
                                                    Location: DBCGLOC

Line Commands:      Commands:
S - Select           Export - Print Results
X - Expand           / - List Commands
/ - List Commands   Ref Object: <TB> TDGF610.TDAT

Cmd Indexes - Creator_Name
-----
<IX> TDGF610.TDAT01CP
<IX> TDGF610.TDAT02NN
    
```

This shows there are 2 indexes available on this table.

Available Commands

The available commands can be shown by using a "/". For example:

```

QBASE Access Paths                                     Lines 1 of 3
Command ==> |                                         Scroll ==> PAGE
                                                    DB2 Subsystem ==> DBCG

S                                     LM SORT SORT
SE L M S B A                         OO NEW COMP
UL P                                     X - Expand
BE L M S B A                         / - List commands
C A IOE N                             S - Select
T N XPQ O TT AT IX MC M F E KE UJOG UJOG
01 01 000 01 T I N 03 0             IS NNNN NNNN
TABLE: TDAT
INDEX: TDAT01CP

Command Actions

Object Type: IX
Object Name: TDGF610.TDAT01CP

Command:
SG Storage Groups SM Statement PL Plans
DB Database L List CL Collection
TS TableSpaces X Expand PK Package
P Partitions S Select EX Explain SQL
TB Tables Y View CK Check Const
IX Indexes E Edit
Vw View DI Display
CO Columns ST Start
AL Aliases SP Stop
TG Triggers VL Volumes
RL Relationships SQ View SQL

Select a Hi-lited choice and press ENTER to process command.
    
```

Additional DB2 SQL Debug Features and Benefits

- Allows an easy prompt for host variable values without errors with respect to data type
- Executes without copying to SPUFI and making error prone changes
- Explains without binding or copying to SPUFI and substituting parameter markers for host variables
- Fully integrated with other TestBase components like DB2 Catalog Guide and View/Edit

Accelerate Your Application

In spite of massive investments of money, time, and resources into critical DB2 for z/OS projects, most still run late and over budget. Overall application quality is poor in spite of strenuous coding and testing efforts.

DB2 SQL Debug is part of SoftBase's TestBase solution that enables customers to code, test, and deploy DB2 for z/OS applications in a fraction of the time it takes now.

About SoftBase

SoftBase is committed to creating a better DB2 development experience. By combining decades of DB2 experience with a set of proven testing and performance tuning tools and an unmatched customer support team, SoftBase delivers – helping application developers and DB2 Administrators create reliable, high quality DB2 applications faster and with ease. With SoftBase you can count on tools that work as promised and a knowledgeable support team available 24/7.