



Using SQL Performance for DB2:
**Identify SQL Tuning Opportunities
thru Object Analysis**

INTRODUCTION

Object Analysis Reports is one feature that differentiates APPTUNE for DB2 from other SQL tuning products – other vendors require a separate product to provide the same information as the Object Analysis report..

In most cases, I/O time is the single largest component of an SQL statement’s elapsed time. We generally focus on fixing SQL Statements with high getpage counts or statements that use tablespace scans. We focus on increasing the number of Matching Columns, strive to reach Index Only access path and apply different techniques to improve the access strategy.

There are other performance metrics to identify potential tuning candidate SQL Statements. For example if you focus on the “Most Used Object” in the system, you likely address the most frequently used SQL statements or programs in the DB2 Subsystem. The advantage of focusing your best tuning techniques on this SQL statement is that even a minor improvement of a few milliseconds or a few getpages can provide a total savings much larger than you would expect. It’s quite possible that the SQL statement might already execute in subseconds or do just a few getpages. But having a better access strategy for that SQL statement that is executed millions of times would save the most CPU. Let’s go in depth for a specific case.

DETAILS...

The Database Analysis section of Object Analysis report is a good place to start. It gives you the most used Databases for the collector interval as shown below. In this example, DBTIB has 80% of the total getpages.

```
ASQEQRPW/I                               View a Report                               LINE 1 OF 10
Command ==>>                               Scroll ==> CSR_

BMCSftwr.SQMSOBJD  --          DATABASE ANALYSIS          --          07/02 13:43:35
Source : DC02-ACTIVE  Intvl : 07/02 08:00 - 07/02 08:59  More:      + >
-----
```

Actions: B-Buffer Pool P-Page Set T-Table D-Database Display

DB2: PR1B Buffer Pool:

Database	BPool	HRatio	Number	%	Getpage	Tot Time	Avg Time	Number	Tot Time	Avg Time
+ DBTIB	96.6 %		870748	78.7%	1:30.197	00.0001	29638	1:25.991	00.0029	
+ DFIDCUST	79.7 %		173572	15.7%	2:18.739	00.0008	33312	2:18.027	00.0041	
+ DFIDUPG2	84.0 %		23616	2.1%	22.6556	00.0010	3772	22.5954	00.0060	
+ DBACH	99.9 %		16812	1.5%	00.2868	00.0000	24	00.1116	00.0047	
+ DFIDUPGR	74.8 %		12772	1.2%	16.7904	00.0013	3220	16.7627	00.0052	
+ DFIDPIOR	99.4 %		7016	0.6%	00.0670	00.0000	44	00.0358	00.0008	
+ DSNDB06	99.8 %		1644	0.1%	00.0734	00.0000	4	00.0250	00.0062	
+ WKDBPR1B	100 %		140	0.0%	00.0004	00.0000	0	00.0000	00.0000	
+ DSNRLST	100 %		8	0.0%	00.0000	00.0000	0	00.0000	00.0000	

F1=Help F2=Split F3=End F4=Sort A F5=Sort D F6=Zoom F7=Up
F8=Down F9=Swap F10=Left F11=Right F12=Cancel

When we drill down one more level to Table Level, the most accessed object is is VP_CUSTOMER_LIMITS table with 485155 getpages.

```

ASQEORPW/I                               View a Report                               LINE 1 OF 15
Command ===>                               Scroll ==> CSR_

BMCSftwr.SQMSOBT --      TABLE ACCESS SUMMARY      --      07/02 13:48:50
Source : DC02-ACTIVE  Intvl : 07/02 08:00 - 07/02 08:59  More: + >
-----
Actions:  T-Table and Index  C-ConnID          R-Program          U-User
          M-Access Matrix    I-CorrID          P-Plan            S-Stmt

DB2: PR1B  Buffer Pool:          Database: DBTIB

Table          BPool  +-----+ Getpage +-----+
Creator  Table Name          Summary  HRatio  Number  Pct  Tot Time  Avg Time
-----
+ SCTIB  VP_CUSTOMER_LIMITS  TAB+IND  95.9 %  485155  55.7%  1:16.8010  00.0002
+ SCTIB  LOGON                TAB+IND  99.1 %   91791  10.5%  0:01.5599  00.0000
+ SCTIB  VP_TRANSACTION_GRO  TAB+IND  100 %   60353   6.9%  0:00.2963  00.0000
+ SCTIB  VP_TRANSACTION_EXT  TAB+IND  92.1 %   57492   6.6%  0:03.0965  00.0001
+ SCTIB  VP_TRANSACTION      TABLE   100 %   49955   5.7%  0:00.3077  00.0000
+ SCTIB  VP_USER              TAB+IND  95.3 %   46571   5.3%  0:01.5759  00.0000
+ SCTIB  VP_CORPORATE         TAB+IND  97.2 %   37648   4.3%  0:00.7800  00.0000
+ SCTIB  VP_TRANSACTION_GRO  TABLE   100 %   28611   3.3%  0:00.1850  00.0000
F1=Help  F2=Split  F3=End    F4=Sort A  F5=Sort D  F6=Zoom    F7=Up
F8=Down  F9=Swap   F10=Left  F11=Right  F12=Cancel

```

The next step is to find the most used SQL Statement and Program on that table. There are 2 options here, using either option (R) or (S) Statement. Either way you end up with the most frequently used SQL statement to target your tuning efforts. I prefer (S)tatement option. In one screen you are able to see both the SQL Statement and Program info at the same time.

```

ASQEORPW/I                               View a Report                               LINE 1 OF 8
Command ===>                               Scroll ==> CSR_

BMCSftwr.SQMSOBT --      OBJECT & SQL STMT ANALYSIS      --      07/02 14:01:59
Source : DC02-ACTIVE  Intvl : 07/02 08:00 - 07/02 08:59  More: + >
-----
Actions:  S-Stmt Stats  H-Header

DB2: PR1B  Buffer Pool:          Database: DBTIB          Page Set:
Plan:          User:          CorrID:          ConnID:
          Creator: SCTIB          Name: VP_CUSTOMER_LIMITS

Program  Stmt  Call  Sect  BPool  +-----+ Getpage +-----+ +-----+
Type    Type  Type  No.  HRatio  Number  %  Tot Time  Avg Time  Number
-----
+ GET_DAIL  STATIC  CURSOR  2  86.3 %  134909  27.8%  1:10.018  00.0005  18417
+ GET_DAIL  STATIC  CURSOR  2  99.7 %  127922  26.4%  01.8657  00.0000  437
+ GET_DAIL  STATIC  CURSOR  2  99.6 %  79709  16.4%  01.2532  00.0000  290
+ GET_DAIL  STATIC  CURSOR  2  99.9 %  76691  15.8%  00.2099  00.0000  41
+ UPD_DAIL  STATIC  UPDATE  2  100 %  28092  5.8%  00.0518  00.0000  0
+ UPD_DAIL  STATIC  UPDATE  2  99.9 %  26288  5.4%  00.1434  00.0000  0
+ INS_DAIL  STATIC  INSERT  2  90.9 %  5780  1.2%  03.2044  00.0006  528
F1=Help  F2=Split  F3=End    F4=Sort A  F5=Sort D  F6=Zoom    F7=Up
F8=Down  F9=Swap   F10=Left  F11=Right  F12=Cancel

```

This is one of the most useful reports for your the tuning process. In one screen, we have the name of the most heavily used program (GET_DAIL) and the most heavily use SQL statement in that program. The SQL Statement mostly accessing table when it's compared with index usage.

SQL Calls:		9288		Statement Type: STATIC		Call Type: CURSOR	
+- IN-SQL	Elapsed +	+- IN-SQL	CPU --+	BPool	+----- Getpage -----		
Tot Time	Avg Time	Tot Time	Avg Time	HRatio	Total	Avg	Tot Time %
+ SCTIB	VP_CUSTOMER_LIMITS	T	BP8K0	86.2 %	130899	14	1:07.221 67.8%
+ SCTIB	IXCSTLIMITS	I	BP0	88.0 %	4010	0	02.7972 2.8%

Before going into the Explain feature we can hypothesize that the Filter Factors of Index columns or the order of key columns are not optimal, the number of Matching Columns is not as high as it could be and the access path is not Index Only. Most of the getpages are against the table. These are just guesses based on the report above. Additionally we can also see that the Index is using BP0 which is normally reserved for the DB2 Catalog and Directory.

If we can decrease the number of table touches by increasing index usage (increase the number of matching columns or index screening) and ideally turn to access path to Index Only, then the average number of getpages might even decrease from 14.

```

BMCSftwr.SQMCACT -- SQL STATEMENT TEXT -- 07/02 14:57:56
Source : DC02-ACTIVE Intvl : 07/02 08:00 - 07/02 08:59 More: - >
-----
Actions for +: T-Detail E-Errors H-Header Q-CatSQL I-HistExpl O-OBJECT
for *: X-Explain SQL text C-Explain Compare
Subsys: PR1B CorrID: Plan: ClntAp:
ConnID: User: ClntID:
AppGrp: ClntWS:

SELECT SCTIB.VP_CUSTOMER_LIMITS.ID, SCTIB.VP_CUSTOMER_LIMITS.DAILY_SUM
FROM SCTIB.VP_CUSTOMER_LIMITS
WHERE SCTIB.VP_CUSTOMER_LIMITS.CUSTOMER_ID = :H :H
AND SCTIB.VP_CUSTOMER_LIMITS.CORPORATE_USER_CUSTOMER_ID IS NULL
AND SCTIB.VP_CUSTOMER_LIMITS.TRANSACTION_ID IS NULL
AND SCTIB.VP_CUSTOMER_LIMITS.TRANSACTION_GROUP_ID = :H :H
AND SCTIB.VP_CUSTOMER_LIMITS.DATE = :H :H FETCH FIRST 1 ROWS ONLY

```

The screen shot shows the SQL statement we need to improve. Upon further examination of the Explain results, the index used for this SQL has only two columns and has 0.12 Cluster ratio. Having "Equal" and "IS" operators in the WHERE clause makes this SQL a perfect candidate for 5 column match and Index Only access. Adding the other 3 columns in the WHERE clause plus two columns in SELECT section into the index makes this an "SQL Index Only" access with less than 10 getpages average.

This is an example of tweaking the most used SQL and producing a saving a few getpages that can makes a huge difference in CPU utilization.

ABOUT THE AUTHOR



Cuneyt Goksu is Principal Information Management Consultant in VBT and has been working with DB2 more than 20 years. He studied Computer Science (CS), holding MBA and MS in CS specialized in Database Modeling.

Cuneyt worked as DB2 for z/OS DBA and DB2 Systems Programmer for 10 years with large scale customers. He also managed large ADABAS, VSAM and Oracle Clusters. Since 2001 he has been working as DB2 SME and Consultant. He focused on DB2 installation and migration, subsystem and application performance & tuning, security health checks, infrastructure design reviews, data and application modeling, dataSharing implementations, modernization and database migrations, Integration and Federation Projects, DB2 Tool Implementations and DB2 Training.

Since he joined with VBT in 2002, Cuneyt is actively working on BMC DB2 Tools Pre-Sales Cycles and Implementation Projects. Cuneyt is a certified IBM solutions expert and holds many IBM Information Management Technical and Sales certifications, active member of the IDUG community since 2003, currently member of IDUG BOD, leader of Turkish DB2 User Group, served as a member of Paper Selection Committee of IBM Information On Demand Europe in 2009&2010, IBM Information Champion and IBM DB2 Gold Consultant. He is also Authorized DB2 Training Partner and member of IBM Academic Initiative Program.

BUSINESS RUNS ON I.T. I.T. RUNS ON BMC SOFTWARE

Business runs better when IT runs at its best. Tens of thousands of IT organizations around the world -- from small and mid-market businesses to the Global 100 -- rely on BMC Software (NASDAQ: BMC) to manage their business services and applications across distributed, mainframe, virtual and cloud environments. BMC helps customers cut costs, reduce risk and achieve business objectives with the broadest choice of IT management solutions, including industry-leading Business Service Management and Cloud Management offerings. For the four fiscal quarters ended September 30, 2012, BMC revenue was approximately \$2.2 billion. www.bmc.com

