

Getting Started with  
**MAINVIEW<sup>®</sup> for DB2**  
and **RxD2<sup>™</sup>**

**MAINVIEW<sup>®</sup> for DB2 6.1**  
**RxD2<sup>™</sup> 2.1**

**Version 6.1**

**May 17, 2000**



Copyright © 1992-2000 BMC Software, Inc., as an unpublished work. All rights reserved.

BMC Software, the BMC Software logos, and all other BMC Software product or service names are registered trademarks or trademarks of BMC Software, Inc. IBM and DB2 are registered trademarks of International Business Machines Corp. All other registered trademarks or trademarks belong to their respective companies.

## Restricted Rights Legend

U.S. GOVERNMENT RESTRICTED RIGHTS. UNPUBLISHED—RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES. Use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth in FAR Section 52.227-14 Alt. III (g)(3), FAR Section 52.227-19, DFARS 252.227-7014 (b), or DFARS 227.7202, as amended from time to time. Send any contract notices to Contractor/Manufacturer:

**BMC Software, Inc.**  
2101 CityWest Blvd.  
Houston TX 77042-2827  
USA

---

## Contacting BMC Software

You can access the BMC Software Web site at <http://www.bmc.com>. From this Web site, you can obtain general information about the company, its products, special events, and career opportunities. For a complete list of all BMC Software offices and locations, go to <http://www.bmc.com/corporate/offices.html>.

### USA and Canada

**Address** BMC Software, Inc.  
2101 CityWest Blvd.  
Houston TX 77042-2827

**Telephone** 713 918 8800 or  
800 841 2031

**Fax** 713 918 8000

### Outside USA and Canada

**Telephone** (01) 713 918 8800

**Fax** (01) 713 918 8000

---

---

## Customer Support

You can obtain technical support by using Response Online™ (comprehensive information from the Web) or Response On Demand™. To expedite your inquiry, please see “Before Contacting BMC Software,” below.

### Response Online

You can obtain technical support from BMC Software 24 hours a day, seven days a week by accessing the technical support Web site at <http://www.bmc.com/support.html>. From this site, you can

- read overviews about support services and programs that BMC Software offers
- find the most current information about BMC Software products
- search a database for problems similar to yours and possible solutions
- order or download product documentation
- report a problem or ask a question
- subscribe to receive e-mail notices when new product versions are released
- find worldwide BMC Software support center locations and contact information, including e-mail addresses, fax numbers, and telephone numbers

### Response On Demand

In the USA and Canada, if you need technical support and do not have access to the Web, call 800 538 1872. Outside the USA and Canada, please contact your local support center or your local sales office for assistance.

### Before Contacting BMC Software

Before you contact BMC Software, have the following information available so that a technical support analyst can begin working on your problem immediately:

- product information
  - product name
  - product version (release number)
  - license number and password (trial or permanent)
- operating-system and environment information
  - machine type
  - operating system type, version, and service pack or program temporary fix (PTF)
  - system hardware configuration
  - serial numbers
  - related software (database, application, and communication) including type, version, and service pack or PTF
- sequence of events leading to the problem
- commands and options that you used
- messages received (and the time and date that you received them)
  - product error messages
  - messages from the operating system, such as `file system full`
  - messages from related software

---

---

# Contents

<b>Chapter 1. Quick Reference Card</b> .....	1
Where to Start .....	3
How to Activate Requests .....	4
<b>Chapter 2. Isolating DB2 Performance Problems</b> .....	5
Enter MAINVIEW .....	6
Check DB2 Status and Activity .....	7
Check Status of All DB2s .....	8
Current Thread Activity .....	11
Exceptions Overview .....	13
Thread Activity for Multiple DB2s .....	13
Analyze One DB2 .....	14
Check DB2 Status .....	15
Buffer Pools .....	17
Lock Contention Analysis .....	19
Page Set Status and I/O Analysis .....	22
Use Monitors to Isolate Problems .....	26
Start a Monitor .....	27
Review Critical Problems .....	30
Exceptions .....	31
Warning Overview .....	32
More Detailed Information .....	33
Windows-Mode Monitor Views .....	35
Workload Objective Views .....	37
Audit Trail .....	38
Issue DB2 Commands .....	40
<b>Chapter 3. Monitoring a DB2 Data Sharing Group</b> .....	41
Define the Group Context .....	42
Check Current Group Activity .....	45
Look at Page Set Considerations (I/O / GBP-DEP) .....	46
Analyze Global Lock Contention .....	50
Tune Group Buffer Pools .....	54
<b>Chapter 4. Tuning an Application with Trace</b> .....	59
Start an Application Trace .....	60
Specify Options .....	61
Activate the Trace .....	66
Review Tips on Tracing Tests .....	67
Look for Application Problems .....	69
All Trace Entries (DB2 Accounting Records) .....	71
Data for One Thread .....	72
View DB2 Accounting Data .....	73
View Detail Event Summaries (Detail Trace Only) .....	76
Another Thread .....	82
Detail Events .....	83
Summarized Data for All Threads .....	85
Application I/O Analysis .....	86
Print a Trace Report .....	88
All Data per Traced Thread .....	89
Other Examples .....	91

<b>Chapter 5. Analyzing the DB2 Workload with Trace</b> .....	93
Analyze Recent Workload History .....	94
Graphic Display of Thread History .....	94
Workload Analysis .....	96
Isolating Problems .....	97
Review Past History .....	100
Current Log Data Set .....	100
Earlier Log Data Set .....	101
Analyze DB2 I/O .....	102
Run a System I/O Trace .....	103
Analyze I/O by Page Set .....	104
Analyze I/O by Application .....	105
Print Workload Reports .....	106
<b>Chapter 6. Using RxD2 with MAINVIEW for DB2</b> .....	111
Improve Performance with SQL Prototyping .....	112
Start a Trace .....	113
Test a Statement .....	114
EXPLAIN a Statement .....	116
Compare Test Results .....	118
Print Results .....	120
Access DB2 Catalog and PLAN_TABLE Information .....	121
Accessing the RxD2 Primary Option Menu .....	121
Accessing EXPLAIN for Currently Executing SQL .....	122
Accessing EXPLAIN from a Trace .....	124
Accessing Other Catalog Data with Direct Hyperlinks .....	128
Accessing Specific Objects in the Catalog .....	130
<b>Chapter 7. Printing Reports</b> .....	135
Print Online Reports .....	136
History Traces (Thread Data) .....	136
All Data per Traced Thread .....	137
Other Accounting Report Examples .....	140
Other Detail Trace Report Examples .....	141
Online Views .....	142
Print Offline Reports .....	143
Batch Trace Print .....	143
Trace Print from a TLDS .....	144
Trace Print from SMF Records .....	147
Performance Reporter Reports .....	148
Reports from SMF .....	148
Reports from DB2 Tables .....	150
<b>Index</b> .....	153

---

## About This Book

This tutorial is intended for first-time users of MAINVIEW for DB2 and RxD2. It takes you step-by-step through practice sessions with several fundamental product applications.

**Note:** You will need appropriate DB2 authorization to execute certain steps.

## Conventions Used in This Book

The following syntax notation is used in this tutorial:

- Items you type and keys you press are highlighted with **bold** letters.
- An item in CAPITAL LETTERS must be entered exactly as shown.
- Items in lowercase letters are values you supply.
- A vertical line | separates alternative options; one must be chosen.

The term pop-up refers to an ISPF pop-up display that replaces the original screen image.



---

## Chapter 1. Quick Reference Card

The following page contains a quick reference card for your use. You can tear it out and keep it by your terminal if you wish.

It contains

- A brief list of where you should start looking when you want information about a certain topic
- Examples for defining requests

**Important**

If you are a new user, be sure to go through each of the practice sessions in the rest of this book before using the quick reference card.



## Quick Reference Card

## MAINVIEW for DB2

### Where to Start

To See	Start Here
DB2 Primary Option Menu	Option 5 on MAINVIEW Selection Menu
Multiple DB2s	EZDSSI Easy Menu
One DB2	EZDB2 Easy Menu
Applications analysis	EZDBA Easy Menu
Tuning wizards	EZDWIZ Easy Menu
All windows-mode views	MAIN (or VIEWS) view
A full-screen-mode service from windows mode	EZDFAST Easy Menu or TRANsfer target product; service  Examples: TRAN DB2P DB2; LOG or TRAN DB2P DB2; EX LOCKD
A windows-mode view from full-screen mode	Option V or TRANsfer target MVDB2; view
Status of all DB2s	STDB2 view
Detail status of one DB2	STDB2D view or DB2ST (Option 1) (full-screen mode)
Current threads	THDACTV view or USERS (full-screen mode)
Current locks	LOCKD or LOCKU (full-screen mode)
Lock contention analysis	Lock Analysis tuning wizard (WZLOCK view)
Data sharing analysis	Data Sharing tuning wizard (WZDSHAR view)
Page sets	EZDPS Easy Menu
I/O analysis	EZDPS Easy Menu or Option 7 (full-screen mode)
Buffer pools	EZDBFRPL Easy Menu
Group buffer pools	EZDSSI / EZDBFRPL Easy Menus or Data Sharing tuning wizard (WZDSHAR view)
Monitors	EZDB2 / EZDSSI Easy Menus
Workload objectives	EZDB2 / EZDSSI Easy Menus
Current traces	Option 4 (full-screen mode)
History traces	Option 5 (full-screen mode)
Recent workload history	Option 6 (full-screen mode)
DB2 catalog information	RxD2 (RX option)

## How to Activate Requests

- **To define requests for automatic startup** (BLKDMRW member in BBPARM):

- Summary Trace example

```
REQ=ATRAC THRDHIST TYPE=SUMMARY TITLE='THREAD HISTORY' STORAGE=4000K
LOGTRAC=Y TRNUMDS=3 TRSWTIME=24:00 TRDSN='Myafx.Trace.Dsn.V01'
```

- Detail Trace example

```
REQ=ATRAC DETLABC DB2PLAN=ABC TYPE=SQL,SCAN,I/O
STORAGE=4000K WRAP=Y TITLE='I/O TRACE OF ABC' GROUPSQL=Y
TRBUFF=5, TRSIZE=800K
```

- Monitor example

```
REQ=BPUTL BPO WMAX=85 I=00:01:00 WLIM=99 LOG=ATWARN
```

See “SET Keyword Parameter Options” in Volume 1 of the *MAINVIEW for DB2 User Guide* for a complete description of all the keywords you can use.

- **To define requests to check active thread exceptions** (“runaway queries”) or other background exception conditions (DMRBEX00 member in BBPARM):

- TSO Exception example

```
MSG=DZ0630W, CPUTOT=1000, GPTOT=100000
```

See sample member DMRBEXBB in BBPARM or “Part 4 - Monitors” in Volume 1 of the *MAINVIEW for DB2 User Guide* for a complete list of all the background monitors.

To display the active background samplers, you can type **BG ON** from the Active Timer Requests application (Option 3).

- **To activate, modify, or purge individual traces**, use the Start Trace panels (ST option from Current Traces (Option 4)).
- **To activate, modify, or purge individual monitors**, use the Start Monitors panels (SM option from Active Timer Requests (Option 3)).
- **To activate a block request with additional monitors or traces manually**, type on any full-screen display:

```
SERV ==> SET
PARM ==> BLK=bl kmbrname
```

where bl kmbrname is BLKDMRW or a user-created block request member in BBPARM.

- **To purge all existing requests from BLKDMRW manually**, type on any full-screen display:

```
SERV ==> SET
PARM ==> PRG=BLKDMRW
```

---

## Chapter 2. Isolating DB2 Performance Problems

These scenarios teach you how to navigate easily through the MAINVIEW for DB2 views and displays and use the available facilities. They do not show you every area covered by the product or all of the displays.

In this practice, session you

1. Check status and activity of all DB2s to detect potential problem areas.
2. Analyze a single DB2 to study problem areas in more detail.
3. Use monitors to isolate specific resource- or workload-related problems.
4. Review critical problems, including those just identified by workload monitors.

See the *Using MAINVIEW* manual for a complete description of how to work in both windows mode and full-screen mode.

**Note:** This practice session takes approximately one hour to complete.

## Enter MAINVIEW

If you are new to MAINVIEW, you should get a copy of the document *Quick Start with MAINVIEW* before starting. This consists of a few stapled pages that cover the basic things you need to know to use any MAINVIEW product. The first page has some blank lines that should be filled in with your startup options. See *Using MAINVIEW*, TD-906, for more detailed information.

Enter MAINVIEW by executing the MAINVIEW CLIST, selecting an ISPF panel option, or logging on to a VTAM session. Then go to Option 0 to specify your CASID, and, when this is done, return to the MAINVIEW Selection Menu, as shown in [Figure 1](#).

*Accessing  
MAINVIEW  
for DB2*

```

----- MAINVIEW Selection Menu -----
OPTION  ==> 5                                DATE  -- 5/11/00
                                           TIME  -- 15:36:58
                                           USERID -- BOLLAA2
                                           MODE  -- ISPF 4.2

  0 Parameters      Specify MAINVIEW options
  1 PLEXMGR         MAINVIEW Plex administration
  2 FOCAL POINT    Subsystem monitoring and alerts
  3 AutoOPERATOR   Automation and resource control
  A MVALARM        MAINVIEW Alarm management
  T InTune         Program analysis and tuning
  V VistaPoint     Comprehensive view of applications and resources

MAINVIEW for
  4 CICS           CICS performance and control
  5 DB2           DB2 performance and control
  6 IMS           IMS performance and control
  7 MVS           OS/390 (MVS) performance and control
  8 MQSeries      MQSeries performance and control
  9 USS           UNIX System Services performance and control
  N Networks      Network performance and optimization

  X EXIT          Terminate MAINVIEW

                                Copyright BMC Software, Inc. 2000
    
```

Figure 1. MAINVIEW Selection Menu

You can enter any installed MAINVIEW product from this menu. Start by selecting Option 5 to access MAINVIEW for DB2.

## Check DB2 Status and Activity

Begin by accessing the MAINVIEW for DB2 Primary Option Menu, as shown in [Figure 2](#).

*Analyzers*  
*Monitors*  
*Traces*  
*Thread History*  
*Views*  
*DB2 Catalog*  
*DB2 Console*

```

BOOLE & BABBAGE ----- PRIMARY OPTION MENU ----- MAINVIEW for DB2 6.1.0
OPTION  ==> V                                     DATE -- 5/11/00
                                                TIME -- 13:45:08
                                                USERID -- CIR11
                                                MODE -- ISPF 4.2

    Managing DB2 Performance:
    1 STATUS          - DB2 Status (DB2ST)
    2 ANALYZERS       - Current Status/Activity Displays
    3 MONITORS        - Early Warnings/Recent History (Active Timer Requests)
    4 TRACES          - Current Application Traces
    5 HISTORY TRACES  - Historical Trace Data Sets
    6 GRAPH           - Recent Thread History
    7 I/O             - DB2 I/O Analysis
    8 BBI INFO        - BBI Subsystem Information
    V VIEWS           - Windows Mode (New Facilities)

DB2 administration:
RX RxD2 FlexTools

General Services:
C CYCLE SETUP       - Service Refresh Cycle Setup
L LOG DISPLAY       - Display Logs
M MESSAGES          - Display Messages and Codes
K KEYS              - Current PF Key Assignments
T TUTORIAL          - Tutorials/News/Getting Started
X EXIT              - Terminate
PF1/13: HELP
PF3/15: EXIT

```

Figure 2. MAINVIEW for DB2 Primary Option Menu

From here, you have direct access to all facilities provided by MAINVIEW for DB2. Most of the options shown here provide direct access to data displays and control panels for managing a single DB2 at a time. You can change the target DB2 directly on almost every screen. Most of these functions operate in full-screen mode.

However we are first going to investigate the capabilities of the windows-mode views, since they are specially designed to provide an overview of multiple DB2 subsystems at a time, as well as drill-down to details about any single DB2.

Point-and-shoot *hyperlinks* take you from one view to other displays with related information. Sometimes these displays are further windows-mode views; sometimes they are full-screen displays normally accessed through these menu options.

## Check Status of All DB2s

To access windows mode and get acquainted with the views available for monitoring multiple DB2s at a time:

1. From the Primary Option Menu, select the **VIEWS** option.

OPTI ON ==> V

The DB2 SSI Easy Menu (EZDSSI) is displayed, as shown in [Figure 3](#).

*Window Information Line ==>*

*DB2 Overview*

```

11MAY00 16:32:36 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
WI =EZDSSI===== (ALL=====*)=====) 11MAY00==16:37:22====M/DB2=====1
                                DB2 SSI Easy Menu
      Status
      . Overview
      . Exceptions
      > Buffer Pools
      > Status Menu

      Data Sharing
      . GBP Group Status
      . GBP Group Activity
      . Global Lock Stats
      . Global Lockouts
      . Volume I/Os (SSI)
      > Page Set Menu

      Monitors
      . In Warning
      . Summary By Area
      . Active
      . Workload Objectives

      Threads
      . Current Threads
      . DB2 Summary
      . Attach Summary
      . Connect Summary
      . Plan Summary

      Tools And Menus
      . Set SSI Context
      > Easy Menu
      > MAIN Menu
      > Tuning Wizards
      . Return...
    
```

Figure 3. DB2 SSI Easy Menu (EZDSSI)

2. Look at the fourth line from the top.

This is the *window information line*. It shows the number of this window (since you may later have several windows open at once), the name of this view (EZDSSI), the current target context (ALL), date, time, and product (MVDB2).

This view is an example of an Easy Menu that provides hyperlinks to various other views.

**Note:** Options prefixed with a period take you directly to data, while options prefixed with a “>” take you to another menu.

This EZDSSI Easy Menu is designed to be used in Single System Image (SSI) mode to look at multiple target DB2s at one time.

3. There are two hyperlinks to other Easy Menus that can be used to check out DB2 status:

- **Status Menu** provides options to select the many views that show complete DB2 statistics (EZDSTAT). EZDSTAT hyperlinks generally go to tabular views that show one row per DB2 with detail views accessible from there.
- **Buffer Pools** provides options to show status and statistics for individual buffer pools (and group buffer pools).

Look at these menus but don’t go further now. You can come back here later.

4. To check the status of all DB2s, put your cursor on the **Overview** hyperlink and press **Enter**.

This takes you to the STDB2 view, as shown in [Figure 4](#), maintaining the default context of ALL, which includes all defined DB2s.

*Check Status of All DB2s*

```

11MAY00 16: 41: 51 ----- INFORMATION DISPLAY -----
COMMAND ==>>
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =STDB2===== (ALL=====*)=====) 11MAY00==16: 41: 51====MVDB2=====3
DB2      Act   Comm  Getpg  Lockout Total          Dataset  Dataset  G
Target   Thrd  Rate   Rate   Rate Excpt  Warning Msg      In- Use  Open
DB0GC    0   0.0    0.0    0.0    0 Connect Fail      0
DB0HC    0   0.0    0.0    0.0    0 Connect Fail      0
DB1GC    0   0.0    0.0    0.0    0 Connect Fail      0
DB1HC    2   0.0    0.1    0.0    0 GBL cont >2%      4      18
DB2GC    0   0.0    0.0    0.0    1 Connect Fail      0
DB2HC    5   0.0   11.0    0.0    0 GBL cont >2%     16     37
    
```

Figure 4. DB2 Activity Overview (STDB2)

With this view, you can quickly gauge how well each of your DB2 subsystems is performing by reviewing and comparing the number of active threads, key activity rates, a total exception count, and the most critical warning condition that exists right now (if any). Notice the “>” sign before the W1 in the information line. This indicates that you can scroll right for more data.

Later you will see how you can move these fields around and set thresholds to customize the view to meet *your* monitoring needs exactly (or even create different views for different conditions).

- Several of the column headers are highlighted, indicating that you can choose a row in that column. This hyperlinks to a view with more detail (related to that column) for the selected DB2.

If you see a warning message for one DB2, place the cursor on that message and press **Enter** to see a list of all the warning conditions that are tracked per DB2. If multiple conditions exist at the same time, only the most important is shown in STDB2, and the others are set to YES in this view. Press **PF3** to return to STDB2.

**Note:** You can do this hyperlink to see the list even if the warning message field is blank (all the conditions will be set to NO).

- Hyperlink on the **DB2 Target** field for an active DB2 to see a detail status view.

You will return here later. Press **PF3** to return to STDB2.

- Hyperlink on the **Comm Rate** field for an active DB2 to see counts and rates of activity in that DB2, both for the current interval (1 to 15 minutes) and for the total session data since DB2 startup.

## Check DB2 Status and Activity

### Interval and Session Counts

```

11MAY00 16:37:54 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1      ALT WIN ==>
>W1 =STDB2====STRATE==(ALL=====DB2HC====) 11MAY00==16:37:52====MVDB2=====1

```

	Interval Qty	Session Qty	Interval Rate	Session Rate	Interval /Thread	Session /Thread	I
Commits....	16	52	0.0	0.0	2.0	1.1	
Thd Creates.	8	46	0.0	0.0			
Getpages...	18431	30434	38.8	1.2	2303.9	661.6	
Page Updates	209	211	0.4	0.0	26.1	4.6	
Sync I/O...	259	560	0.5	0.0	32.4	12.2	
Prefetch I/O	2153	3502	4.5	0.1	269.1	76.1	
Write I/O...	4	5	0.0	0.0	0.5	0.1	
DatasetOpens	16	45	0.0	0.0	2.0	1.0	
Lockouts...	2	2	0.0	0.0	0.2	0.0	
LockSuspend	21	51	0.0	0.0	2.6	1.1	
GBLLockCont.	36	924	0.1	0.0	4.5	20.1	
Clm/Drn Fls.	0	0	0.0	0.0	0.0	0.0	
DML SQL....	1196	1700	2.5	0.1	149.5	37.0	
StProcCalls.	0	0	0.0	0.0	0.0	0.0	
StProcFails.	0	0	0.0	0.0	0.0	0.0	
RID Fails...	0	0	0.0	0.0	0.0	0.0	
EDM Loads...	5	10	0.0	0.0	0.6	0.2	
EDM Fails...	0	0	0.0	0.0	0.0	0.0	
Checkpoints.	0	1	0.0	0.0	0.0	0.0	
Prll Groups.	0	0	0.0	0.0	0.0	0.0	
PrllFallback	0	0	0.0	0.0	0.0	0.0	

Figure 5. Activity Rates (STRATE)

Again, some of these fields are highlighted, indicating that hyperlinks will take you to even more detail.

8. Press **PF3** to return to STDB2.

## Current Thread Activity

From the DB2 status overview (STDB2), you can access current thread information:

1. Hyperlink on **ACTV THRD** to view a list of all active threads in the selected DB2, as shown in [Figure 6](#).

*All Active Threads*

```

11MAY00 11: 59: 34 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> CSR
CURR WIN ==> 1           ALT WIN ==>
>W1 =THDACTV=====DB2H=====*=====11MAY00==11: 59: 20====MVDB2=====3
      DB2 Connect           Elapsed           SQL
Auth ID ID Name CPU Time Time %CPU Stmt Warn Locks User
BOLHHH4 DB2H DB2CALL 00: 01: 22.06 00: 08: 24.48 2.1 2 No 8 ACTIV
BOLLAA2 DB2H DB2CALL 00: 01: 07.16 00: 05: 58.88 0.8 23251 No 15 ACTIV
BOLLAA2 DB2H DB2CALL 00: 00: 00.00 00: 00: 02.11 0.0 2 No 4 SWAPP
DMRC DB2H DB2CALL 00: 00: 32.12 05: 40: 25.95 0.0 0 No 0 BBI M
    
```

Figure 6. All Active Threads for One DB2 (THDACTV)

2. Type **SORT D** on the COMMAND line, move your cursor to the **Elapsed Time** header, and press **Enter** to sort the threads by the length of time they have been active.
3. Hyperlink on one thread by placing your cursor in the **AUTH ID** column to see a detail display for that thread, as shown in [Figure 7](#).

*Detail User*

*Expand to  
Locks Held =>  
SQL Detail =>*

```

BOOLE AND BABBAGE----- DETAIL USER STATUS ----- RX AVAILABLE
SERV ==> DUSER           INPUT 15: 01: 07 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> BOLHHH4           ROW 1 OF 80 SCROLL=> CSR
EXPAND: MON(USER), UTRAC, ST(START TRACE), LOCKE, EXPLAIN, PT, CICSE, CMRTASK
ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, SPAS, DDF
CURRENT..... 11: 07: 13.53 PLAN..... DSNESPRR TYPE..... ALLIED
START..... 10: 58: 49.39 AUTHI D..... BOLHHH4 CONNECT..... TSO/TSO
ELAPSED..... 00: 08: 24 ORIG PRIM AUTH. .... BOLHHH4 CORR ID. .... BOLHHH4
STATUS..... IN-DB2 COMMI TS..... 0 ROLLBACKS..... 0
-----
RUNTIME ANALYSIS IN DB2 IN APPL. TOTAL %IN DB2(=) TOTAL(*)
----- 0...25...50...75...100%
ELAPSED TIME 00: 08: 23 718 ms 00: 08: 24 |=====*|
CPU TIME 00: 01: 22 123 ms 00: 01: 22 |===|
DB2 WAIT TIME 20 s |<|
- - - - - ACTIVITY - - - - - KEY INDI CATORS - - - - -
TOTAL SQL..... 2 SQL: DYNAMI C(PREPARE)= 1
GETPAGES..... 416, 803 I/O RSP: SYNC= 45 ms, ASYNC= 107 ms
SYNC READS (PRL=00) ..... 147
PREFETCH PAGES READ..... 1, 051
UPDATES/COMMI T..... 0. 0
BFR HIT RATIOS: ... VP=100%, HP=100%
-----
-SQL STATEMENT ANALYSIS - - - - -
STATEMENT #: 193 STATEMENT TYPE: OPEN
ACTIVE IN DB2 CPU TIME: 1, 897 ms ELAPSED TIME: 00: 01: 06
Package/DBRM: DSNESM68 (DYNAMI C) PLAN ISOLATION LEVEL: RR
PROCEDURE/UDF/TRIGGER: DSN8SPAS_TEST
SELECT * FROM LONG_SQL ;
    
```

*Expand to  
SQL Detail =>*

*SQL Statement*

Figure 7. Detail User Status Display (DUSER)—Base Section

## Check DB2 Status and Activity

This display shows all available accounting detail information about that thread. The most critical data is summarized in the base section, including key indicators of failures or potential problems. You can scroll down, use the expand buttons, or point-and-shoot to view complete detail data. Also, the DB2 thread cancel command can be invoked with a CANCEL parameter if you have the proper authorization.

4. The full text of the current SQL statement being executed is available near the bottom of this base section.

If your window size is too small to show the full text, position the cursor on the first text line and scroll down with **PF8**.

Notice that this section also includes the current package/DBRM name and the time this SQL statement has been active.

5. Tab to the **SQLCOUNTS** button or to the **TOTAL SQL** line and press **Enter** to view a breakdown of all SQL executed by this thread.

6. Press **PF3** once to return to the DUSER base section.

7. If this is an active IMS or CICS thread and if MAINVIEW for IMS or MAINVIEW for CICS is installed, the detail display will include an expand button to expand directly to that product:

- MAINVIEW for CICS: TASKXPND display for this CICS transaction
- MAINVIEW for IMS: DREGN display for the region processing this IMS transaction

Press **PF3** once to return to DUSER.

8. There are many other sections of detail data with which you will get acquainted over time. For example, if you are concerned about locking, you could choose the **LOCKE** expand button to see all the locks held (or waited on!) by this thread and any threads in contention with it.

9. Press **PF3** several times to return to EZDSSI.

## Exceptions Overview

Another way to check for exception conditions in any of your DB2s quickly is to choose the **Exceptions** hyperlink on EZDSSI to bring up the view STEXC, as shown in [Figure 8](#).

### Exception Conditions

```

11MAY00 16: 52: 20 ----- INFORMATION DISPLAY -----
COMMAND ===>                                     SCROLL ===> CSR
CURR WIN ===> 1          ALT WIN ===>
W1 =STEXC===== (ALL=====*) 11MAY00==16: 50: 58====MVDB2=====4
DB2      Warning System      User
Target   Monitors Exceptions Exceptions Warning Msg
DB1GC           0           0           0 Connect Fail
DB1HC           0           0           0 GBL cont >2%
DB2GC           0           0           0 Connect Fail
DB2HC           1           0           0 GBL cont >2%

```

Figure 8. Exception Conditions (STEXC)

This view identifies immediately whether or not any exceptions have been detected, and shows what kind they are. Hyperlinks lead to additional details.

**Warning Monitors** show conditions that have been detected by timer-driven sampling monitors defined with warning thresholds. Typically, a predefined group of monitors is started automatically and it is always active. A later section provides more information on monitors.

**System Exceptions** are conditions that are detected by a background sampler and are not governed by a threshold; for example, a problem with log archiving.

**User Exceptions** are thread-related conditions detected by a background sampler. Threshold conditions can be defined per attach type; for example, IMS, CICS, or batch.

The first three types generate warning messages that can be viewed while active, tracked historically in the journal log, or fed into an automated operator product like MAINVIEW AutoOPERATOR.

**Warning Msg** shows the most important of several conditions that are checked per interval, and on request, as part of the total DB2 status and statistics data collection for windows-mode views. These conditions may also be identified by one of the other exception samplers.

1. If you have a non-zero value for any of these counts, follow the hyperlinks to see further details.
2. When finished, press **PF3** as needed to return to EZDSSI.

## Thread Activity for Multiple DB2s

From EZDSSI, you can choose hyperlinks to view thread activity across multiple DB2s:

1. Hyperlink on **Current Threads** to see all active threads, identified by the DB2 they are running in.
2. Hyperlink on **DB2 Summary** to see a summary of threads for each DB2; then hyperlink on **one row** to view a list of threads for just that DB2.

## Analyze One DB2

From EZDSSI, choose the **Easy Menu** hyperlink to access an Easy Menu designed to analyze one DB2 at a time, EZDB2, as shown in [Figure 9](#). This selects one DB2 from the context of ALL.

*One DB2*

```

11MAY00 12:41:15 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =EZDB2=====DB2HC====*=====11MAY00==12:41:14====MVDB2=====1
                                DB2 Easy Menu
      DB2 Target --->  DB2HC
      Status
      . Overview
      . Exceptions
      . Buffer Pools
      . DDF Connections
      . Activity Rates
      > Status Menu

      Lockouts
      . Events
      . Resource Summary
      . Connection Summary
      . Blocker/Waiter Summary

      Page Sets
      . Page Set Status
      . Volume I/O Summary
      > Page Set Menu

      Monitors
      . In Warning
      . Summary By Area
      . Active
      . Workload Objectives

      Tools And Menus
      . Set Target Context
      . DB2 Catalog/Explain
      > SSI Easy Menu
      > DBA Easy Menu
      > Fast Menu
      > MAIN Menu
      > Tuning Wizards
      . Return...
  
```

Figure 9. DB2 Easy Menu (EZDB2)

To select the target DB2 you want to analyze:

1. Hyperlink on **Set Target Context** to see a list of targets; then place your cursor in the **Target** column and press **Enter** to hyperlink to a different target.

**Note:** If you already know the DB2 target you want, simply type **CON target** on the COMMAND line.

2. Notice in the **Tools And Menus** section, there are hyperlinks to RxD2 (DB2 Catalog/Explain) as well as several other Easy Menus.

## Check DB2 Status

To check the status of the target DB2:

1. Hyperlink on **Overview**.

This takes you to the STDB2 view, as shown in [Figure 4 on page 9](#), but only the one DB2 is displayed.

2. Now you may want to review recent history.

Type **TIME \* \* 2H** to see the last 2 hours by 15-minute intervals. Type **INCLUDE TIME** to see the times (the Intvl Time column is added to the view), as shown in [Figure 10](#).

*Review History*

```

11MAY00 16: 57: 55 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =STDB2=====DB2HC=====*=====11MAY00==16: 57: 54====MVDB2=====1
DB2      Intvl Act  Comm  Getpg  Lockout Total          Dataset  Data
Target   Time  Thrd  Rate  Rate   Rate Excpt Warning Msg    In-Use  0
DB2HC      5    0.0  4.7   0.0   0  GBL cont >2%      4      63
    
```

Figure 10. DB2 Activity Overview (STDB2)

You can hyperlink on a time period for further analysis if desired. (Don't do it now though.)

3. Reset to current time with **TIME \* \* 1I** and then **EXCLUDE TIME**.
4. Hyperlink on **DB2 Target** to see a detail status view, as shown in [Figure 11](#).

*Detail Status View*

```

11MAY00 16: 58: 31 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =STDB2=====STDB2D===DB2HC=====*=====11MAY00==16: 57: 54====MVDB2=====1
> EZ Menu      . Sessi on
                Connect  Active      In DB2      Queued      Suspended    CPU%
TSO.....      1          1           0           0           0           0.0
Batch.....    4          4           2           0           1          32.0
-CAF.....    4          4           2           0           1          32.0
-Utility...   0          0           0           0           0           0.0
IMS.....     0          0           0           0           0           0.0
CICS.....     0          0           0           0           0           0.0
DBAT.....     0          0           0           0           0           0.0
SPAS.....     9          0           0           0           0           0.0
RRSAF.....    0          0           0           0           0           0.0
*Total*....   14         5           2           0           1          32.0

                0... 50. 100          0... 50. 100
GBL cont >2%  EDM Pool          Total CPU%

BP Rates:..  Getpage      4.7      Reads      0      Writes      0.0
Locking:..  Suspend      0      Deadlocks  0      Timeouts   0
Exceptions: Monitor    0      System     0      User       0
Data Sets:. Open        63      Open HWM   63      In-use     4
Parallel:.. Maxinum     0      Groups     0      Fall back  0
STOPROCS:.. Calls      0      Abends     0      Timeouts   0
Group BP:.. Reads      2      Writes     2      Failures   2
Paging:.... DB2         0.0     System     0.0
DB2 Start:.. Date      11MAY00 Time      09: 19: 46
    
```

Figure 11. DB2 Status Detail—Interval (STDB2D)

This shows an overview of current thread activity per attach type as well as key indicators of recent performance.

5. Use the **.Session** hyperlink at the top of the view to see totals since DB2 startup instead of current interval values in the key indicator fields.
6. The **EZ Menu** option takes you to EZDSTATD.

This Easy Menu is very similar to the DB2 Status Easy Menu, EZDSTAT, that you saw earlier, but the hyperlinks here take you directly to all the detailed statistics views for this DB2.

7. Hyperlink on **SQL Counts** to see STSQLD as an example of these detailed statistics views, as shown in [Figure 12](#).

*Detailed View*

```

11MAY00 17:01:51 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1      ALT WIN ==>
WI =STDB2====STSQLD====DB2HC====*=====11MAY00==16:57:54====MVDB2=====1
                                Interval      Sessi on
Data Manipulation Language
SELECT.....                0          0
INSERT.....                 0          0
UPDATE.....                 2          6
DELETE.....                 0          0
DESCRIBE.....               0          0
PREPARE.....                12         205
OPEN.....                   10         199
FETCH.....                  150        46998
CLOSE.....                   9          197
Data Defini ti on Language..
CREATE TABLE.....          0          0
INDEX.....                  0          0
    
```

Figure 12. SQL Counts Detail (STSQLD)

8. Press **PF3** to return to EZDSTATD.  
You may want to browse a few other detail views.
9. Press **PF3** to return to STDB2D.
10. Hyperlink on **TSO** to see a tabular view of current TSO users.
11. We have stepped through several status views now. However, some information is not available in windows mode. Examples are information about CICS connections, DB2 logging, DSNZPARM values, and so forth.

To access these displays, simply press **PF3** to return to the Primary Option Menu, or you can use one of the following quick paths:

- From EZDB2, choose the **Fast Menu** to get a menu of hyperlinks into full-screen mode.
- Use the transfer command at any point to access the full-screen DB2 Status display DB2ST. This is similar to STDB2D, but it provides point-and-shoot to most full-screen displays, such as CICSC (to see CICS connections), ZPARM, and so forth.

**TRAN target DB2; EX DB2ST**

You can also use the transfer command to access other MAINVIEW products.

## Buffer Pools

To analyze buffer pools for the target DB2:

1. From EZDB2, hyperlink on **Buffer Pools** to see a list of all defined pools with allocated space, utilization, and getpage rates, as shown in [Figure 13](#).

### All Defined Pools

```

11MAY00 17: 04: 18 ----- INFORMATION DISPLAY -----
COMMAND ==>> SCROLL ==>> CSR
CURR WIN ==>> 1 ALT WIN ==>>
W1 =BFRPL=====DB2HC====*=====11MAY00==17: 04: 18====MVDB2=====9
Pool DB2 Vpool Vpool Hpool Hpool Getpg Get- % % Active
Name Target Size Alloc Size Alloc /sec pages Actv 0.... 50.... 100
BP0 DB2HC 200 200 0 143.2 37236 0
BP1 DB2HC 25 0 0.0 0 0
BP2 DB2HC 25 30 0.0 0 0
BP4 DB2HC 21 0 0.0 0 0
BP5 DB2HC 30 0 0.0 0 0
BP9 DB2HC 20 0 0.0 0 0
BP11 DB2HC 400 0 0.0 0 0
BP32K DB2HC 20 0 0.0 0 0
BP32K9 DB2HC 20 0 0.0 0 0

```

Figure 13. Buffer Pool Statistics—SSI (BFRPL)

2. Hyperlink on **Pool Name** to see all status and statistics for one pool.

Both interval and session counts are shown, as shown in [Figure 14](#).

### Statistics for One Pool

```

11MAY00 17: 05: 05 ----- INFORMATION DISPLAY -----
COMMAND ==>> SCROLL ==>> CSR
CURR WIN ==>> 1 ALT WIN ==>>
W1 =BFRPL====BFRPLD====DB2HC====*=====11MAY00==17: 04: 18====MVDB2=====1

Pool Name..... BPO
VP Size..... 200
Hyperpool Size..... 0
Virtual Pool Sequential Threshold..... 80
Hyperpool Sequential Threshold..... 80
Deferred Write Threshold..... 50
Vertical Write Threshold..... 10
Castout Attribute Y/N..... Y
Parallel Sequential Threshold..... 50
Assisting Parallel Threshold..... 0

Interval Sessi on
Virtual Pool Allocation.....
  Buffers Allocated..... 200
  Current Active Buffers..... 1
  Expansions/contractions..... 0 0
  No Buffer - Pool Full..... 0 0
  Expansion Fail..... 0 0
Hyperpool Allocation.....
  Expanded Storage Buffers..... 0
  Buffers Allocated..... 0
  Expansions/Contractions..... 0 0
Virtual Pool Read Statistics.....
  Getpages..... 37236 135607
  Sequential Getpages..... 36910 128765
  Sync I/O..... 104 1657
  Sequential Sync I/O..... 0 289

```

Figure 14. Local Buffer Pool Statistics Detail (BFRPLD)

3. Scroll down with **PF8** to see all the data. Press **PF3** to return to BFRPL.
4. Hyperlink on **Getpg/sec** to see current hit ratios per pool (BFRPLH); then hyperlink on **Pool Name** to see both current and session ratios (BFRPLHD) for one pool, as shown in [Figure 15](#).

*Hit Ratios*

```

11MAY00 17:07:12 ----- INFORMATION DISPLAY -----
COMMAND ==>                                SCROLL ==> CSR
CURR WIN ==> 1          ALT WIN ==>
W1 =BFRPL====BFRPLHD==DB2HC====*=====11MAY00==17:04:18====MVDB2=====1
Pool Name..... BPO

                Interval   Sessi on
Hit Ratios.....
VPOOL Hit Ratio % with P/F...   17.51   18.55
VPOOL Hit Ratio % without P/F   92.64   81.70
GBP Hit Ratio %.....           0.00   18.92
Activity per Second.....
Getpages.....                143.20   4.87
Page Updates.....             29.78   0.61
Sync I/O.....                  0.40   0.06
Prefetch I/O.....             15.51   0.51
Write I/O.....                  0.80   0.02
Data Set Opens.....           0.00   0.00
    
```

Figure 15. Buffer Pool Rates Detail (BFRPLHD)

5. Press **PF3** to return to EZDB2.

Group buffer pools are discussed later in the data sharing section (see [“Tune Group Buffer Pools”](#) on page 54).

## Lock Contention Analysis

Although locking problems must usually be resolved at the application level, the first indication of problems—and the simplest identification of the applications and resources involved—can best be seen at the system level. To use the options in the Lockouts section of EZDB2 to look at locking problems in one DB2 (without data sharing):

1. If you aren't sure whether or not you are having much lock contention, you may want to review the lockout and suspension counts and rates first in the STRATE view (hyperlink on **Activity Rates**).

A certain number of lock suspensions (and even an occasional timeout or deadlock) is to be expected in an active DB2 system. More frequent lockout conditions (either timeouts or deadlocks) may be a cause for concern, since the SQL statements involved are terminated.

2. You can hyperlink to see a list of the last 100 timeouts and deadlocks either from the **Lockouts** field in STRATE, or from **Lockout Events** on EZDB2.

Both go to the view LKEVENT, as shown in [Figure 16](#).

### Timeouts and Deadlocks

```

11MAY00 17: 20: 11 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                     SCROLL ==>> CSR
CURR WIN ==>> 1           ALT WIN ==>>
>WI =LKEVENT=====DB2HC====*=====11MAY00==17: 20: 10====MVDB2=====5
Date / Time   Lockout  Victim  Victim  Victim  Victim  Victim  Nr.
----- Type   Plan    AuthID  Connect CType  Corr ID  Res.
11MAY- 16: 58: 56 DEADLOCK  AUDIT  BLLAT5  TSO     TSO     BLLAT5  2
11MAY- 16: 58: 45 TIMEOUT  PAYROLL DMRDLK3 BATCH  BATCH  DMRDLK3  1
11MAY- 16: 42: 34 TIMEOUT  PAYROLL DMRTMO3 BATCH  BATCH  DMRTMO3  1
11MAY- 16: 37: 33 DEADLOCK  AUDIT  BLLAT5  TSO     TSO     BLLAT5  2
11MAY- 16: 36: 53 TIMEOUT  RXDB2  AXCNM1 1 DB2CALL CAF     AXCNM1 1

```

Figure 16. Lockout Events (LKEVENT)

This view shows a chronological list of up to 100 lockout events that have occurred since DB2 startup, sorted with the most current at the top of the screen.

3. Type **SORT** and tab to the **Victim Plan** column to sort these lockout events by plan name to identify the applications for which SQL requests were terminated.
4. Hyperlink on the **Date/Time** field for an event (preferably a deadlock that involves multiple resources).

The first view shows the resources involved and holder (blocker) and waiter plans.

5. Scroll right to see additional details for both resources at once, or hyperlink on the **Res Seq** field to see all the available details for this conflict.
6. Press **PF3** to return to EZDB2.

Although you may have analyzed lockout conditions one at a time like this before, the windows-mode views provide additional analysis capabilities. They are available through the other three hyperlinks in the lockouts section of EZDB2.

- Hyperlink on **Resource Summary** to do an analysis of the resources involved in timeouts or deadlocks, as shown in [Figure 17](#).

*Resources Involved*

```

11MAY00 17: 21: 09 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =LKRESZ=====DB2HC====*=====11MAY00==17: 21: 09====MVDB2=====2
--Resource Name-- Total          % Total          Global
Database Object Conflicts      0.. 50.. 100  PAGE  ROW INDEX Conflicts
DSN8D51A DSN8S51E          5  71.4  *****          5    0    0          2
DSN8D51A DSN8S51P          2  28.6  ***              0    2    0          2
    
```

Figure 17. Lockout Resource Summary (LKRESZ)

The first view, LKRESZ, summarizes all conflicts by resource name, usually database and table space. Note that a deadlock with three resources and participants will result in three conflicts, not one, for the purposes of this analysis. With this view, you can easily identify those table spaces involved in the most contention.

- Hyperlink on a **resource name** showing one or more conflicts to see a breakdown of these conflicts by specific resource, down to a page or row level (LKRESNRZ).

With this view, hot spots in your tables are immediately visible, as shown in [Figure 18](#).

*See Hot Spots in Tables*

```

11MAY00 17: 22: 35 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =LKRESZ===LKRESNRZ=DB2HC====*=====11MAY00==17: 22: 35====MVDB2=====1
--Resource Name-- Resource Resource Total          % Total          Global
Database Object Number Type Conflicts      0.. 50.. 100 Conflicts
DSN8D51A DSN8S51E 0000001200 DATAPAGE          5  71.4  *****          2
    
```

Figure 18. Lockout Resource Number Summary (LKRESNRZ)

- Hyperlink on either the **resource name** or the **resource number** to view a list of each lockout event that involved this resource (LKRESD), as shown in [Figure 19](#).

The events are initially sorted in descending sequence by time, but the SORT command can be used to sort by any column. (If you have forgotten how, type **HELP SORT** on the COMMAND line.) This allows you to identify the applications (blocker planname, waiter planname) and users (blocker/waiter Corr ID, Connection, Victim Auth ID) involved quickly.

*Each Event for This Resource*

```

11MAY00 17: 24: 34 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =LKRESZ===LKRESD===DB2HC====*=====11MAY00==17: 24: 34====MVDB2=====5
--Resource Name-- Resource Resource Time Lockout Blocker Waiter Gbl
Database Object Number Type ----- Type Pl anName Pl anName Con
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16: 58: 56 DEADLOCK PAYROLL AUDIT Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16: 58: 45 TIMEOUT AUDIT PAYROLL
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16: 42: 34 TIMEOUT AUDIT PAYROLL
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16: 37: 33 DEADLOCK PAYROLL AUDIT Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16: 36: 53 TIMEOUT RXDB2 RXDB2
    
```

Figure 19. Lockout Resource Conflict Detail (LKRESD)

Although the solution to locking problems may involve application or table redesign, at least you now know where the problems lie.

10. Press **PF3** until you return to EZDB2 and hyperlink on **Connection Summary** if you suspect that the problem may lie in application scheduling, where applications with incompatible lock usage are running concurrently, as shown in [Figure 20](#).

### Application Scheduling Problems

```

11MAY00 17:38:42 ----- INFORMATION DISPLAY -----
COMMAND ==>> SCROLL ==>> CSR
CURR WIN ==>> 1 ALT WIN ==>>
>W1 =LKCONZ=====DB2HC=====11MAY00==17:38:41====MVDB2=====1
DB2      Victim  Victim  Blocker  Timeouts  Deadlocks  % Lockouts
Target   ConnType  Connect  Connect
DB2HC   TSO      TSO      BATCH    0          2  40.0 *****
DB2HC   BATCH   BATCH   TSO      2          0  40.0 *****
DB2HC   CAF     DB2CALL CAF      1          0  20.0 ***

```

Figure 20. Lockout Connection Summary (LKCONZ)

The view LKCONZ summarizes the conflict data to identify, for example, whether batch jobs or utilities are blocking critical CICS or IMS transactions. As in the resource summaries, hyperlinks lead to lists of the exact events with time stamps, so that the critical time periods can be seen at a glance.

11. Press **PF3** until you return to EZDB2 and hyperlink on **Blocker/Waiter Summary** in order to identify incompatible applications that are frequently blocking each other, as shown in [Figure 21](#).

### Incompatible Applications

```

11MAY00 17:40:24 ----- INFORMATION DISPLAY -----
COMMAND ==>> SCROLL ==>> CSR
CURR WIN ==>> 1 ALT WIN ==>>
>W1 =LKBWZ=====DB2HC=====11MAY00==17:40:24====MVDB2=====1
Blocker  Waiter  Timeout  Deadlock  % Conflicts  Global  Participant
PlanName PlanName  Invol v.  Invol v.  . . . .  0 . . . 50 . . . 100  Conf1.  -Only Count
AUDIT    PAYROLL    2          0  40.0 *****  0          0
PAYROLL  AUDIT      0          2  40.0 *****  0          0
RXDB2    RXDB2      1          0  20.0 ***     0          0

```

Figure 21. Lockout Blocker/Waiter Summary (LKBWZ)

The view LKBWZ summarizes the conflict data by blocker and waiter plans, so that you can quickly see which applications are causing the most conflicts. The hyperlinks again show all of the lockout events where the selected plan was involved.

12. Press **PF3** until you return to EZDB2.
13. To see current contention, hyperlink to the **Fast Menu** (on the right under Tools And Menus).

In the Lock Activity section, there are two valuable options:

- Hyperlink on **User Contention** to see a list of active threads with lock counts (LOCKU). Holding and waiting threads in current conflicts are identified.
- Hyperlink on **DB/TS Contention** to see a list of resources in contention, showing the lock owner and waiters (LOCKD).

From both of these displays, you can hyperlink on one conflict to see additional details. The LOCKE display shows a thread with all locked resources and identifies waiter threads per resource.

**Note:** LOCKU is also accessible through hyperlinks in several thread views (THDxxx).

## Page Set Status and I/O Analysis

The tuning of physical I/O and page set usage in the buffer pools is a critical success factor in DB2. DB2 itself uses many techniques to reduce or defer I/O, but the setup, from DASD volumes to buffer pool allocation and thresholds, all plays a part. To use the options in the Page Sets section of EZDB2 and the Page Set Easy Menu itself to look at data object issues with the focus still on a single DB2:

1. The first thing to review is simply the status of all open page sets. Hyperlink on **Page Set Status** to see a list sorted by object name (database, table space, partition), as shown in [Figure 22](#).

Sort by  
Object Name

```

11MAY00 16: 11: 08 ----- INFORMATION DISPLAY -----
COMMAND ==>>
CURR WIN ==>> 1 ALT WIN ==>>
W1 =PSSTAT=====DB2H=====*=====11MAY00==16: 11: 08====MVDB2=====77
-----Page Set----- Bfrpl % First Defr GBP
Database Object Prt ID Ty Users Size(K) Used EXT Volume WrtQ ERR Dep
DB2HWORK DSN4K01 001 BPO0 TS 0 7440 99.4 3 BAB310 0 N
DSNDB01 DBD01 001 BPO0 TS 0 1440 100.0 1 BAB310 1 N
DSNDB01 DSNLLX01 001 BPO0 IX 0 288 66.7 1 BAB310 0 N
DSNDB01 DSNLLX02 001 BPO0 IX 0 240 80.0 1 BAB312 0 N
DSNDB01 DSNLUX01 001 BPO0 IX 0 48 33.3 1 BAB320 0 N
DSNDB01 DSNLUX02 001 BPO0 IX 0 48 33.3 1 BAB320 0 N
DSNDB01 DSN SCT02 001 BPO0 IX 0 144 11.1 1 BAB310 0 N
DSNDB01 DSN SPT01 001 BPO0 IX 0 240 40.0 1 BAB312 0 N
DSNDB01 DSN SPT02 001 BPO0 IX 0 432 44.4 1 BAB325 0 N
DSNDB01 SCT02 001 BPO0 TS 0 10080 14.3 1 BAB310 1 N
DSNDB01 SPT01 001 BPO0 TS 0 5760 25.0 1 BAB312 1 N
DSNDB01 SYSLGRNX 001 BPO0 TS 0 1440 100.0 1 BAB310 1 N
DSNDB01 SYSUTLX 001 BPO0 TS 0 1440 100.0 1 BAB320 1 N
DSNDB06 DSNADH01 001 BPO0 IX 0 48 33.3 1 BAB312 0 N
DSNDB06 DSNAGH01 001 BPO0 IX 0 96 16.7 1 BAB314 0 N
DSNDB06 DSNAPH01 001 BPO0 IX 0 96 16.7 1 BAB310 0 N
DSNDB06 DSNATX01 001 BPO0 IX 0 144 11.1 1 BAB312 0 N
DSNDB06 DSNATX02 001 BPO0 IX 0 480 20.0 1 BAB312 0 N
DSNDB06 DSNATX03 001 BPO0 IX 0 432 22.2 1 BAB312 0 N
DSNDB06 DSNAUH01 001 BPO0 IX 0 96 16.7 1 BAB310 0 N
DSNDB06 DSNDCX01 001 BPO0 IX 0 2160 66.7 1 BAB312 0 N
DSNDB06 DSNDDH01 001 BPO0 IX 0 48 33.3 1 BAB310 0 N
DSNDB06 DSNDDX02 001 BPO0 IX 0 48 33.3 1 BAB310 0 N
DSNDB06 DSNDKX01 001 BPO0 IX 0 192 50.0 1 BAB318 0 N
DSNDB06 DSNDLX01 001 BPO0 IX 0 48 33.3 1 BAB312 0 N
DSNDB06 DSNDPX01 001 BPO0 IX 0 96 100.0 1 BAB310 0 N
    
```

Figure 22. Page Set Status (PSSTAT)

2. Sort on the **EXT** column to identify the data sets with the highest number of extents (a potential performance impact). Type **SORT D** and tab to the **EXT** column.

3. If you want to limit the view by selecting only certain page sets, one simple way is to use the **WHERE** command. It works very much like the SQL **WHERE** clause. First choose one (or more) column you want to filter the view on. Place the cursor in that column header and press **PF1** to view the field help. Within the help, the element name is identified.

Now type **WHERE** on the **COMMAND** line. Under Where Condition, type the filters you want in effect.

For example, to show only those data sets with several extents, type

```
I0_EXT > 5
```

Press **PF3** to return to **PSSTAT** with the **WHERE** clause in effect. You can check the filters in effect in any view with the **SHOWFILT** command.

**Note:** Help for any command is available by typing **HELP commandname** on the **COMMAND** line.

As another example of filtering, to see only catalog tables, type

```
I0_DBTSP = DSNDB06*
```

4. Hyperlink on the **Page Set** field to see complete details about the selected page set, including size, volume, buffer pool cache data, and detailed I/O counts and elapsed wait times. You may need to scroll down with **PF8** to see all the data.
5. Press **PF3** to return to **PSSTAT**.
- Hyperlink on the **Users** column to see the threads currently accessing this page set. Or hyperlink on the **ERR** column to see if a restricted status is in effect for this table space.
6. Press **PF3** to return to **EZDB2**.
7. One of the most important issues to check periodically is data set placement and volume I/O response times.

Hyperlink on **Volume I/O Summary** to see the volumes in use for DB2 databases, as shown in [Figure 23](#).

### Volumes In Use

```

11MAY00 11: 21: 48 ----- INFORMATION DISPLAY -----
COMMAND ==>>
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =PSVOLSZ=====DB2H=====*=====11MAY00==11: 21: 48====MVDB2=====11
  DB2      Nr.    Sync  I/O Sync Max  Sync Avg          Async
Volume Target PSs   I/0s  % I/0 Wait I/0 Wait 0. . . 20. . . 40 I/0s
BAB309 DB2H    1      9    2.0    35    19 *****          1
BAB310 DB2H   16    141  31.0   325    24 *****          15
BAB312 DB2H   21    144  31.6   175    24 *****          43
BAB314 DB2H    5     25   5.5   667    41 *****+          12
BAB316 DB2H    4     14   3.1   38     14 ****              0
BAB318 DB2H    9     50  11.0   73     22 *****          21
BAB319 DB2H    1      2   0.4    31     17 *****          0
BAB320 DB2H    5     24   5.3    51     19 *****          6
BAB321 DB2H    1      3   0.7    32     22 *****          0
BAB325 DB2H   13     40   8.8    47     20 *****          7
BAB330 DB2H    1      3   0.7    40     20 *****          1

```

Figure 23. Volume I/O Summary—Session (PSVOLSZ)

8. Type **SORT D** and tab to **Sync Avg I/O Wait** to sort the volumes with the highest average delays to the top.

9. These values are based on activity since DB2 startup.

To see a 2-hour summary of activity for one volume broken down into 15-minute intervals, hyperlink on **Sync I/Os**.

Press **PF3** to return.

10. Hyperlink on a **volume** to see a list of all page sets on that volume.

You may want to scroll to the right to see the asynchronous I/Os made for prefetch, since these I/Os have different access characteristics and delay times than do synchronous I/Os.

11. Press **PF3** until you return to EZDB2 and hyperlink on the **Page Set Menu** option to go to EZDPS.

Hyperlink on any of the **I/O by Page Set** options to look at I/O counts and wait times per page set.

- Sort on the **Avg I/O Wait** column to identify the highest average delays that can point out DASD response time problems.
- Sort on **Max I/O Wait** to help identify occasional contention problems that are masked in the averages.

**Note:** However, be aware that the maximum is since DB2 startup, not per interval. This does reduce its usefulness.

12. Press **PF3** until you return to EZDPS and hyperlink on **Cache Statistics** to analyze page usage in the buffer pools and hiperpools by page set, as shown in [Figure 24](#).

*Cache Statistics*

```

11MAY00 16: 18: 48 ----- INFORMATION DISPLAY -----
COMMAND ==>> SCROLL ==>> CSR
CURR WIN ==>> 1 ALT WIN ==>>
>W1 =PSCACHE=====DB2H=====*=====11MAY00==16: 18: 48====MVDB2=====77
-----Page Set----- Bfrpl VP VP VP VP VPTot VPTot
Database Object Prt ID Current Maximum Changed Max Chng % All %BP
DB2HWORK DSN4K01 001 BP00 0 2 0 2 0.0 0.0
DSNDB01 DBD01 001 BP00 7 23 0 6 4.0 3.5
DSNDB01 DSNLLX01 001 BP00 9 10 0 6 5.2 4.5
DSNDB01 DSNLLX02 001 BP00 3 8 0 6 1.7 1.5
DSNDB01 DSNLUX01 001 BP00 3 4 0 2 1.7 1.5
DSNDB01 DSNLUX02 001 BP00 2 4 0 2 1.2 1.0
DSNDB01 DSNST02 001 BP00 0 3 0 1 0.0 0.0
DSNDB01 DSNST01 001 BP00 0 3 0 1 0.0 0.0
DSNDB01 DSNST02 001 BP00 0 4 0 1 0.0 0.0
DSNDB01 SCT02 001 BP00 1 5 0 3 0.6 0.5
DSNDB01 SPT01 001 BP00 1 5 0 3 0.6 0.5
DSNDB01 SYSLGRNX 001 BP00 8 13 0 7 4.6 4.0
DSNDB01 SYSUTLX 001 BP00 3 12 0 5 1.7 1.5
DSNDB06 DSNADH01 001 BP00 2 3 0 0 1.2 1.0
DSNDB06 DSNAGH01 001 BP00 2 3 0 0 1.2 1.0
DSNDB06 DSNAPH01 001 BP00 0 3 0 0 0.0 0.0
DSNDB06 DSNATX01 001 BP00 2 3 0 2 1.2 1.0
DSNDB06 DSNATX02 001 BP00 6 6 0 2 3.5 3.0
DSNDB06 DSNATX03 001 BP00 3 4 0 2 1.7 1.5
DSNDB06 DSNAUH01 001 BP00 2 3 0 0 1.2 1.0
DSNDB06 DSNDCX01 001 BP00 7 7 0 4 4.0 3.5
DSNDB06 DSNDDH01 001 BP00 2 3 0 0 1.2 1.0
DSNDB06 DSNDDX02 001 BP00 0 3 0 0 0.0 0.0
DSNDB06 DSNDKX01 001 BP00 2 3 0 1 1.2 1.0
DSNDB06 DSNLX01 001 BP00 2 3 0 1 1.2 1.0
DSNDB06 DSNPX01 001 BP00 2 3 0 1 1.2 1.0
    
```

Figure 24. Page Set Cache (PSCACHE)

You may want to sort on the **VP Current** column (descending) to sort those with the highest current storage usage to the top. The **VP Changed** and **VP Max Chng** columns identify those page sets with update activity.

- Press **PF3** until you return to EZDPS and hyperlink on **Summary by Buffer Pool** for assistance in balancing table space allocations to the proper buffer pools, as shown in [Figure 25](#).

*Balance  
Table Space  
Allocations*

```

11MAY2000 16: 21: 03 ----- INFORMATION DISPLAY -----
COMMAND ==>>
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =PSBPSZ=====DB2H=====11MAY2000==16: 21: 01====MVDB2=====1
Bfrpl  DB2      Nr.    VP      VP      HP      Total  VP Max  VPMaxChg
  ID   Target  PSs  Current Changed  Current  I/Os   (1 PSet) (1 PSet)
BP00  DB2H      77    173     0       0      1341   23      15
BP02  DB2H      23    12      3       0       522   84       8

```

Figure 25. Buffer Pool Page Set Summary—Session (PSBPSZ)

The view PSBPSZ gives you a quick overview of how all the buffer pools are being used.

- Hyperlink on Bfrpl ID to see a list of all page sets allocated to that pool (PSBPS). Sort on the **VP Current** column (descending) to sort those with the highest current storage usage to the top.

The **VP Changed** and **VP Max Chng** columns identify those page sets with update activity.

- Press **PF3** until you return to PSBPSZ. Hyperlink on **VP Current** for a 2-hour history.
- Press **PF3** until you return to EZDB2.

## Use Monitors to Isolate Problems

Monitors sample key measurements over time and save short-term history. When thresholds are specified, the measured values are compared and automatic warnings generated.

A default set of monitors defined in BBPARM member BLKDMRW is automatically started. This set can be tailored for each DB2 system with different monitors or different thresholds.

To view the active monitors:

1. From the Primary Option Menu, select the **MONITORS** option.

OPTION ==> 3 (Active Timer Requests)

All *active* monitors are listed here. The current measurement values are shown and plotted compared to the thresholds. The W in the middle marks the warning threshold values. All monitors with acceptable values remain on the left side of the W. Those in warning status extend to the right and are highlighted, as shown in [Figure 26](#).

*Warning Status*

BOOLE AND BABBAGE		ACTIVE TIMER REQUESTS		PERFORMANCE MGMT	
COMMAND ==>		INPUT INTVL ==> 3		TGT ==> DB2G	
				TIME -- 10:52:10	
COMMANDS: SM (START MONITORS), SORT, AREA, X ON OFF, DM (DMON), DW (DWARN)					
LC CMDS: S (SELECT), W (SHOW), M(MODIFY),					
P (PURGE), R (REPLICATE), H (HELP), Z (STOP) >>>					
LC	SERV PARM	TITLE	CURRENT	WVAL	-8-6-4-2-0+2+4+6+8+
	CONUT TSO	CONNECTION % UTILIZATION	35	80	<<<<<<<< W
	CONUT BATCH	CONNECTION % UTILIZATION	2	80	** W
	CONUT DBAT	CONNECTION % UTILIZATION	18	80	>>>> W
	THDUT	THREAD % UTILIZATION	50	85	<<<<<<< W
	THDQD CICS	QUEUED THREADS	0	NZ	W
	THDWT	CREATE THREAD WAITED	0	5	W
	EDMPL	EDM POOL % UTILIZATION	89	85	>>>>>>> W>>>
	BPURL BPO	BFR POOL % UTILIZATION	76	85	***** W
	BPUSE BPO	BFR POOL % IN USE	33	85	*** W
	RIDUT	RID POOL % UTILIZATION	45	85	<<<<< W
	LOGUT	ACTIVE LOG % UTILIZATION	53	85	>>>>> W
	CSAP	CSA % UTILIZATION	95	80	>>>>>>> W>>>>>
	ECSAP	ECSA % UTILIZATION	79	80	***** W
	DSUTL	OPEN DB DATASET % UTIL	24	85	** W
	DSOPN	DB DATA SETS OPEN	38	60	>>>>> W
	DB2DP DBAS	DEMAND PAGING	600	1800	<<< W
	LTIME	LOCK TIMEOUT FAILURES	1	5	** W
	LDEAD	LOCK DEADLOCK FAILURES	0	NZ	W
	EDMLD DBD	AVG EDM REQUESTS/LOAD	0	140	W
	RWP	READS WITH PAGING	0	NZ	W
	WWP	WRITES WITH PAGING	0	NZ	W
	GETPG BPO	GETPAGE REQUESTS	1144	1000	***** W**

Figure 26. Active Timer Requests Application

2. Type **X ON** on the COMMAND line to view only those monitors in exception status.

We'll investigate exceptions more thoroughly later.

3. To find out how to start a monitor (and generate an exception to view):

COMMAND ==> SM (Start Monitor)

This application displays a list of all available monitor services. (SM is an *application transfer* command that saves your current position and transfers you to another application. You can type a transfer command in any COMMAND line or SERV field.)

## Start a Monitor

Monitors look at either resources/system activity (resource monitors) or at workload activity (workload monitors). One workload monitor can be started multiple times to look at different workloads to help isolate specific workload-related problems. Some resource monitors also have parameters to support multiple requests—for example, buffer pool number.

There are many available monitors. You can scroll through the list with PF7/8, sort on any of the columns, or select only those monitors for the area in which you are interested.

To practice starting a monitor:

1. Type **AREA WKLD** on the COMMAND line.

All workload monitor names start with # (for a count) or with an @ (for an average).

2. Type **S** in the line command column next to #SQLM to select a workload monitor.

By selecting #SQLM, you can view some immediate activity because this service measures the number of DML statements issued.

This takes you to the data entry panel, as shown in [Figure 27](#).

### Using Defaults

```

BOOLE AND BABBAGE --- START DB2 WORKLOAD MONITOR REQUEST --- PERFORMANCE MGMT
COMMAND ==>                                         TGT ==> DB2G

                                     #SQLM - DATA MANIPULATIVE SQL

PARM      ==>                                     (Workload Monitor Identifier)
INTERVAL  ==> 00: 01: 00  START ==>                STOP ==>                QIS ==> YES
WWAL      ==>                WMSG ==>            WLIM ==> 10  WIF ==> 1    WIN ==> 1
TITLE     ==>                                     (Title)
RST       ==> HOT                                (Restart Option: HOT, COLD, PUR, QIS)
PLOTMAX   ==>                                     (Maximum PLOT X-Axis Value)
RANGES    ==>                                     (1-4 Range Distr. Upper Limits)
LOG        ==>                                     (NO, ATSTOP, ATPD, ATINTVL, ATWARN)

Specify Selection Criteria:
ICHECK    ==> NO                                (Check elapsed versus Interval time)
CONNTYPE  ==>                                     (TSO, IMS, CICS, BATCH, CAF or blank)
DB2PLAN   ==>
DB2AUTH   ==>
DB2CONN   ==>
DB2CORR   ==>
DB2LOC    ==>

```

Figure 27. Start DB2 Workload Monitor Data Entry Panel

## Use Monitors to Isolate Problems

3. Define optional workload monitor data collection parameters on the data entry panel using mostly defaults.

- a. **PARM ==> id**

You can specify any name as an ID if you want several requests for this monitor to look at different workloads.

- b. **INTERVAL ==>**

Use the default (00:01:00) of one minute data collection intervals.

- c. **START ==>**

Use the default to start at the next even minute, or type a future time in the format of hh:mm:ss.

**Note:** The monitors are synchronized to the even minute to make it easier for you to compare the history values they collect.

- d. **STOP ==>**

Use the default of no stop time, or type a time at which you want the monitor to stop in the format of hh:mm:ss, or the number of intervals (minutes).

- e. **WVAL ==> warning threshold value    Maximum (n) or Minimum (<n)**

A warning message is issued automatically when a sampling exceeds the monitor threshold. Type a low value so you can see a warning. For example, if you expect 100 statements in a minute, type 50. It can be changed to a more appropriate value later.

- f. **WMSG ==>**

If left blank, a warning message is sent to the Journal log when sampling exceeds the monitor threshold (also triggers AutoOPERATOR EXECs). Current warnings are shown on DB2EX. Specify **WTO** to also send a message to the MVS console, or a **TSO id** to send a message to that user.

- g. Optional fields: (For this exercise, accept the defaults)

WLIM	Limit number of warnings sent
WIF	Wait n intervals before issuing a warning
WIN	Wait n intervals before sending nth warning
RST	Monitor reaction if DB2 stops and is restarted
PLOTMAX	Specifies maximum value of X Axis on monitor plot display
RANGES	Limits for range distribution statistics on plot display
LOG	A hardcopy plot is recorded in the BBI-SS PAS Image log

4. Specify workload selection criteria.

**DB2CONN ==> name, name, name**

Optional. Type one or more names, where name can be TSO, BATCH, DB2CALL, an IMSID, or a CICS jobname. Leave blank for total workload.

5. Press **Enter** to start the monitor and return to the list of monitors.

6. Access the Active Timer Requests (AT) display to make more requests.

**COMMAND====> AT**    (transfer to Option 3 - MONITORS)  
or    **====> =3**    (a quick jump without saving a return path)

7. Type **R** in the line command column next to an active workload monitor to replicate an active monitor request.

This replicates the request parameters.

8. Make another request by typing a unique ID in the PARM field and specifying different selection criteria. Press **PF1** (HELP) for an explanation of the values that can be specified, as shown in Figure 28.

### Getting Help

```

BOOLE AND BABBAGE ---- SET KEYWORDS - WORKLOAD SELECTION CRITERIA ---- TUTORIAL

The following keywords can be used to qualify data collection by a workload
monitor:

CONNTYPE   TSO|IMS|CICS|BATCH|CAF|blank
            IMSMPP|IMSBMP|IMSTBMP|IMSCTL|DLI|SYSSERV|APLSERV|UTIL (DMR 3.1 and
            above)
            These keywords allow workload selection by connection type.
            Blank is the default, which selects all the connection types.

ICHECK     YES|NO
            If YES is specified, the transaction is discarded if the elapsed
            time is greater than the requested interval time. NO is the
            default.

            Data is only available at transaction termination time. YES
            prevents long running transactions from skewing the interval
            statistics.

                                                    continued next page...

```

Figure 28. Sample Help Panel

9. Press **Enter** to start the monitor and return to Active Timer Requests.
10. Move the cursor to the COMMAND line and continue to press **Enter** until the new monitors show an ACTIVE status and current measurement values. (Remember, they were synchronized to start at an even minute.)
11. Check that at least one monitor is in warning status (line highlighted).
12. Use the Modify line command to view the monitor options in effect and to modify any that are preceded by an arrow.

```

LC   (Line Command)
M   (for Modify)

```

Choose an active monitor and change its threshold value.

```

WWAL ==> n      (Maximum)
WWAL ==> <n     (Minimum)

```

If you did not start this monitor, you need special authorization to modify or purge it.

13. Press **PF3** to return to the AT panel.

Look at how the graph of the current value compared to the new threshold has changed.

**Note:** To stop a monitor, use the **Z** line command.

You now know how to start and modify monitors. The next dialog shows you how to look at the data.

---

### Review Critical Problems

To review critical problems, including those just identified by monitors, return to the **DB2EX** analyzer service.

- Type

```
OPTI ON ===> 2      (from the Primary Option Menu)
COMMAND ===> AN     (transfer to ANALYZERS)
or    ===> =2      (j ump to Option 2)
```

Then type **S** in the line command column next to the DB2EX service.

- An even quicker way, if you remember the name of the display you want, is

```
COMMAND ===> EXEC DB2EX
```

**Note:** If you want a display with one or more parameters, type **EXEC serv parm1, parm2** on the COMMAND line.  
For example: **EXEC BFRPL BPO**.

- In a service display, just overwrite **SERV ===> DB2EX** and clear any PARM values.

**Note:** To print any display while you are looking at it, use **PF4** for screen print. To print a series of displays while you are viewing them, specify **LOG=Y** in line 2.

## Exceptions

To review the information on the exception panel:

1. Move the cursor to a message and press **Enter**.

A detailed explanation of the message is shown by the MESSAGES general service.

2. Press **PF3** to return to the DB2EX display.

The exception display lists the time the exception occurred and the severity (S for severe, W for warning, I for informational, M for monitor messages). You can enter the severity code in the PARM field to select the messages for display by severity level, as shown in [Figure 29](#).

*Severe  
Messages*

TIME	LEV	MSGNO	EXCEPTION
11: 24: 00	S	DZ1090S	FINAL ACTIVE LOG DATASET 75% FULL
11: 24: 00	S	DZ1110S	BSDS REDUCED TO SINGLE MODE
11: 23: 00	S	DZ1020S	IMS TASK(S) QUEUED FOR THREAD(S)
11: 20: 00	S	DZ1050S	BP(0) DM CRITICAL THRESHOLD REACHED

Figure 29. DB2 Exceptions Panel

**Note:** You have already learned how to change monitor thresholds. The background exception messages (S, W, I) are user-modifiable for each DB2 system independently in BBPARM member DMRBEX00. This is also where you specify thresholds for runaway query exceptions per connection type (TSO, CICS, and so on). These exceptions can be triggered based on CPU percent, GETPAGEs, CPU total, elapsed time, or updates per commit.

## Warning Overview

To return to the list of active monitors (Active Timer Requests - AT) for an overview of important system measurements:

1. Type

SERV ==> =3 (transfer to Option 3 - MONITORS)

Where warning thresholds are specified, the current value is plotted compared to the threshold (W marker). Arrow signs indicate an increasing (>) or decreasing (<) trend. (Yellow if increasing.)

If the threshold is exceeded, that monitor line is highlighted (red) and a corresponding warning message is shown in the Exceptions display (DB2EX).

Usually, a standard set of monitors is started automatically to collect data whenever the target DB2 is up. Look at all monitors already in warning status or approaching it. For example, if both LOCK TIMEOUT FAILURES and THREAD PERCENT UTILIZATION are high, as shown in Figure 30, the lock contentions are probably caused by a high level of user activity. If only the timeouts are in warning status, a single batch job may be holding too many locks (analyzer LOCKU shows details).

**Lock  
Contention**  
==>

BOOLE AND BABBAGE		ACTIVE TIMER REQUESTS		PERFORMANCE MGMT	
COMMAND ==>		INPUT INTVL ==> 3		TGT ==> DB2G	
				TIME -- 10:52:10	
COMMANDS: SM (START MONITORS), SORT, AREA, X ON OFF, DM (DMON), DW (DWARN)					
LC CMDS: S (SELECT), W (SHOW), M(MODIFY),					
P (PURGE), R (REPLICATE), H (HELP), Z (STOP) >>>					
LC	SERV PARM	TITLE	CURRENT	WAL	-8-6-4-2-0+2+4+6+8+
	CONUT TSO	CONNECTION % UTILIZATION	35	80	<<<<<<< W
	CONUT BATCH	CONNECTION % UTILIZATION	2	80	** W
	CONUT DBAT	CONNECTION % UTILIZATION	18	80	>>>> W
	THDUT	THREAD % UTILIZATION	89	85	>>>>>>> W >>>
	THDQD CICS	QUEUED THREADS	0	NZ	W
	THDWT	CREATE THREAD WAITED	0	5	W
	EDMPL	EDM POOL % UTILIZATION	50	85	<<<<<<< W
	BPURL BPO	BFR POOL % UTILIZATION	76	85	***** W
	BPUSE BPO	BFR POOL % IN USE	33	85	*** W
	RIDUT	RID POOL % UTILIZATION	45	85	<<<<< W
	LOGUT	ACTIVE LOG % UTILIZATION	53	85	>>>>>> W
	CSAP	CSA % UTILIZATION	72	80	>>>>>>> W
	ECSAP	ECSA % UTILIZATION	80	80	***** W
	DSUTL	OPEN DB DATASET % UTIL	24	85	** W
	DSOPN	DB DATA SETS OPEN	38	60	>>>>> W
	DB2DP DBAS	DEMAND PAGING	600	1800	<<< W
	LTIME	LOCK TIMEOUT FAILURES	8	NZ	***** W *****
	LDEAD	LOCK DEADLOCK FAILURES	0	NZ	W
	EDMLD DBD	AVG EDM REQUESTS/LOAD	0	140	W
	RWP	READS WITH PAGING	0	NZ	W
	WWP	WRITES WITH PAGING	0	NZ	W

Figure 30. Active Timer Requests Application

## More Detailed Information

To view more detailed information:

1. If you are interested in one specific area, such as buffer pools, you can narrow down the display to show only the related monitors.

COMMAND ==> **AREA BUFR** (enter AREA alone to see them all again)

This makes it easier to concentrate on just the buffer information; for example, to compare the GETPAGE REQUESTS to READ I/O and PREFETCH READ I/O.

2. If you don't see the data you want, you can use the same technique to review the other available monitors for buffer pool information.

COMMAND ==> **SM** (Option 2 - START MONITORS, with stacking)

COMMAND => **AREA BUFR**

You could start additional buffer pool monitors now, just the way you did for #SQLM.

3. Press **PF3** to return to the list of active monitors.

4. Use the Select line command to view the history collected by a monitor.

LC (Line Command)

S (for Select)

Choose an active monitor that is showing non-zero for the current value. If the DB2 you are monitoring is very active, there should be several, including the #SQLM monitor you just started. If not, choose one of the MVS-related monitors such as CSAP, CSA Percent Utilization.

This standard **PLOT** display, shown in [Figure 31](#), is used to show the latest 10 values measured by any monitor. For further trending information, the averages for the total monitoring period and the last 2 periods (10 samples each) are also shown. The threshold value is indicated with a W in the graph. Note the **MAXimum** value ever measured and the time it occurred (MAX AT). Rate per second calculations (in this example, paging rates) are shown at the right.

*Rates/Second*

*Maximum Value*

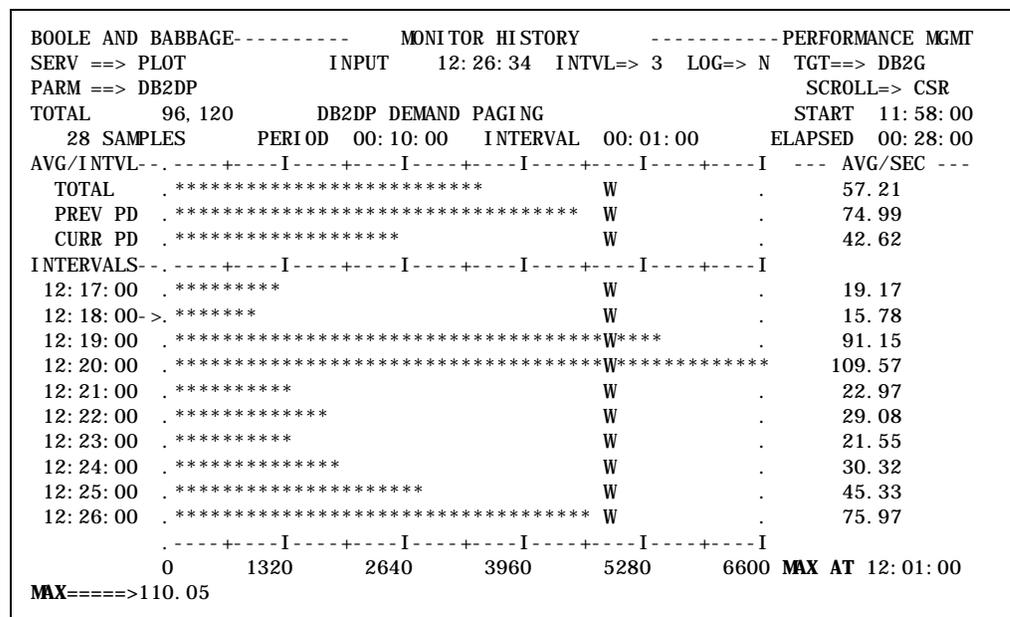


Figure 31. Sample PLOT Display

5. Press **PF3** to return to the list of active monitors.

6. Restrict the display to monitor exceptions only.

COMMAND ===> **X ON**

7. Press **PF6** to start automatic refresh mode, or type

COMMAND ===> **GO**

Check to see that the word **INPUT** on line 3 has been replaced by **RUNNING**.

You can start automatic refresh whenever you want to watch for problems or events without continually pressing **Enter**.

8. Press **ATTN** to stop refresh and enable command entry.

9. Return to the full display.

COMMAND ===> **X OFF**

10. Change to some other installed MAINVIEW product.

COMMAND ===> **product name**

IMS, CICS, DB2, AO, CAO, IAO, and MAO are valid names.

This saves your current place and presents the Primary Option Menu of that product. You can even stack copies of MAINVIEW for DB2 by typing **DB2**. This can be useful when you must look at another system or product (for example, to answer a user question about production), but you don't want to lose your place.

11. Return to your original place (Active Timer Requests).

COMMAND ===> **=X**

You can also press **PF3** in the stacked environment until you exit from its Primary Option Menu.

## Windows-Mode Monitor Views

All of the monitor data you have just seen in full-screen displays is also available in windows mode. Both the EZDB2 and EZDSSI Easy Menus have several hyperlinks to monitor views, as shown in [Figure 32](#).

*Monitor Views ==>*

```

11MAY00 16:32:36 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> CSR
CURR WIN ==> 1          ALT WIN ==>
W1 =EZDSSI===== (ALL=====*) 11MAY00==16:37:22====MVDB2=====1
                                DB2 SSI Easy Menu

  Status
  . Overview
  . Exceptions
  > Buffer Pools
  > Status Menu

                                +-----+
                                | Place cursor on |
                                | menu item and  |
                                | press ENTER   |
                                +-----+

  Threads
  . Current Threads
  . DB2 Summary
  . Attach Summary
  . Connect Summary
  . Plan Summary

  Data Sharing
  . GBP Group Status
  . GBP Group Activity
  . Global Lock Stats
  . Global Lockouts
  . Volume I/Os (SSI)
  > Page Set Menu

  Monitors
  . In Warning
  . Summary By Area
  . Active
  . Workload Objectives

  Tools And Menus
  . Set SSI Context
  > Easy Menu
  > MAIN Menu
  > Tuning Wizards
  . Return...

```

Figure 32. DB2 SSI Easy Menu (EZDSSI)

The monitor views are able to provide an SSI perspective on monitor status, as well as the details for one DB2. To see these views:

1. Check to be sure that you have an SSI context in effect. If not, type

**CON ALL**

2. Hyperlink on **Monitors—In Warning** to see a view of all monitors in warning status in that SSI context, as shown in [Figure 33](#).

*See All Warnings*

```

11MAY00 11:59:40 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> CSR
CURR WIN ==> 1          ALT WIN ==>
W1 =DMWARN=====DB2HC=====11MAY00==11:59:39====MVDB2=====1
CMD Serv Parm          % Warning  Curr   Warn   Area Target
-----
                                0. . . . 50. . . 100 Value-- Value--
@ELTM ALLWORK 319.3 *****+      3.19   1.00 WKLD DB2HC
CSAP          136.0 *****+      68.00   50.00 DMVS DB2HC

```

Figure 33. Monitors in Warning (DMWARN)

- Hyperlink on the **Serv** column to see the equivalent of the full-screen PLOT display, as shown in [Figure 34](#).

Notice that the time intervals here have the most recent times at the top. Each plot view has a hyperlink to a related full-screen display.

*Monitor Plot  
in Windows  
Mode*

```

11MAY00 12:01:55 ----- INFORMATION DISPLAY -----
COMMAND ==> 1 SCROLL ==> CSR
CURR WIN ==> 1 ALT WIN ==>
>W1 =DMWARN===D@ELTM===DB2HC====*=====11MAY00==12:00:58====MVDB2=====1
>>USERS |.....|.....| Elapsed Events Parm.... ALLWORK
12:00:30 66.108 3 Appl.... SAMPLE
11:59:00 0.331 6 Warni ng.. 1.00
11:57:30 0.000 0 Max/Mi n.. Maxi mum
11:56:00 0.000 0 Val ue... 2304.72
11:54:30 0.000 0 Ti me.... 11:48:30
11:53:00 0.000 0 Graph Max 2304.72
11:51:30 16.022 1 Target... DB2HC
11:50:00 0.000 0 Descript. Sample DB2 Workload
11:48:30 ***** 2304.721 2
11:47:00 2.330 1 Samp les.. 69
|.....|.....| Elapsed Events Period... 00:15:00
Total * 86.226 69 Samp Int. 00:01:30
Prev Pd 0.000 0 Start.... 10:17:00
Curr Pd 3.361 28 Elapsed.. 01:43:30
    
```

Figure 34. Sample Plot View

- Press **PF3** twice to return to EZDSSI.
- Hyperlink on **Monitors—Summary by Area** to see a summary of active monitors and their warning status by target and area, as shown in [Figure 35](#).

*See Monitors  
by Target  
and Area*

```

11MAY00 18:05:18 ----- INFORMATION DISPLAY -----
COMMAND ==> 1 SCROLL ==> CSR
CURR WIN ==> 1 ALT WIN ==>
W1 =DMAREAZ===== (ALL=====*)===== 11MAY00==18:05:16====MVDB2=====10
CMD          Number Number          Average Maximum Number
--- Target   Area   Monitors in Warn 0..... 10 Warning Warning Active
DB2GC        DMVS      2         0           74.1   95.2     2
DB2HC        EDM        2         0           40.0   40.0     2
DB2HC        DMVS      5         1 *          38.4   136.0    5
DB2HC        LOG        8         0           11.8   23.5     8
DB2HC        LOCK       8         0            1.6    8.0      8
DB2HC        DSYS      4         0            1.5    2.4      4
DB2HC        USER     22         0            0.8    3.7     22
DB2HC        DDF       3         0              3       3         3
DB2HC        WKLD     6         1 *          319.3  319.3    6
DB2HC        BUFR    24         0              24       24       24
    
```

Figure 35. Monitor Summary by Area (DMAREAZ)

The hyperlinks provide a list of the monitors.

- Press **PF3** to return to EZDSSI.
- Hyperlink on **Monitors—Active** to see a summary of active monitors per target DB2.

Additional views are available to show the monitor data from a realtime (DMONR) or session (DMONS) perspective, while DMONC shows all time perspectives in one view.

- Press **PF3** to return to EZDSSI.

## Workload Objective Views

Special workload monitors are started automatically for you to provide service-level monitoring of response time per workload.

Several default workloads have been predefined for transaction and query connections to DB2, based on connection type. Please refer to Volume 3 of the *MAINVIEW for DB2 User Guide* if you want to add workloads or modify the response time objectives of the existing workloads.

To see if you are meeting your response time goals for the defined workloads:

1. From EZDSSI, hyperlink on **Monitors—Workload Objectives** to review workload objectives, as shown in [Figure 36](#).

*Are You  
Meeting  
Your Goals?*

```

11MAY00 11: 59: 40 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                     SCROLL ==>> CSR
CURR WIN ==>> 1          ALT WIN ==>>
W1 =DOBJ===== (ALL=====*)=====) 11MAY00==14: 49: 33====MVDB2=====1
CMD Workload      % <= Resp Goal Resp  Goal Avg      Tran  Composite Target
--- Name-----  0 . . . . 50 . . 100 Goal -   % Resp---  Count Workload- -----
  ALLWORK   95.6 *****          1.00  95  0.843   125 SAMPLE  DB2H
  ALLWORK   66.7 *****          1.00  95  1.428   167 SAMPLE  DB1H
  CAF       100.0 *****          5.00  90  3.221     2 DB2SAMP  DB1H
  CICS      100.0 *****          1.00  95  0.543   160 DB2SAMP  DB1H
  DBATDB2   00.0                      5.00  90  0.000     0 DB2SAMP  DB1H
  DBATDRDA  00.0                      5.00  90  0.000     0 DB2SAMP  DB1H
  IMSMPP    00.0                      1.00  95  0.000     0 DB2SAMP  DB1H
  TSO       52.1 *****          5.00  90  7.331     5 DB2SAMP  DB1H

```

Figure 36. Objectives Review (DOBJ)

The graph shows what percentage of all threads in that workload have met the response time goal.

## Audit Trail

To view an audit trail of problems throughout the day:

1. Transfer to the Log Display general service (Option L).

COMMAND ==> **LOG**

Or press **PF5**.

All *MAINVIEW for DB2 messages* are shown chronologically, as shown in [Figure 37](#).

*Look for  
Problems*

Boole and Babbage	-----	Log Display	-----	General services
COMMAND ==>				TGT ==> DB2G
LINE=	12, 340	LOG= #1	STATUS= INPUT	TIME= 17: 51: 38
				INTV==> 3
12: 11: 00	DS0560W	(04)	12: 11: 00	ECSA % UTILIZATI ON(TOTAL) = 71 (>70) *****
12: 12: 00	DS0560W	(05)	12: 12: 00	ECSA % UTILIZATI ON(TOTAL) = 71 (>70) *****
12: 12: 55	XS6311I		BBI /SESSION FOR -CPS17 - TERMINATED	
12: 13: 00	DS0560W	(06)	12: 13: 00	ECSA % UTILIZATI ON(TOTAL) = 71 (>70) *****
12: 14: 00	DS0560W	(07)	12: 14: 00	ECSA % UTILIZATI ON(TOTAL) = 71 (>70) *****
12: 15: 00	DS0560W	(08)	12: 15: 00	ECSA % UTILIZATI ON(TOTAL) = 72 (>70) *****
12: 16: 00	DS0560W	(09)	12: 16: 00	ECSA % UTILIZATI ON(TOTAL) = 71 (>70) *****
12: 17: 00	DS0560W	(10)	12: 17: 00	ECSA % UTILIZATI ON(TOTAL) = 71 (>70) *****
12: 22: 11	XS6304I		BBI /SESSION FOR -LAA1 - TO -D31X- INITIATED	
13: 12: 00	DS0561I		13: 12: 00	ECSA % UTILIZATI ON(TOTAL) NO LONGER > 70
13: 28: 48	DSNW131I		- STOP TRACE SUCCESSFUL FOR TRACE NUMBER(S) 05	
13: 28: 49	DSN9022I		- DSNWVCM1 '-STOP TRACE' NORMAL COMPLETION	
13: 53: 02	DS0560W	(01)	13: 53: 00	ECSA % UTILIZATI ON(TOTAL) = 72 (>70) *****
13: 54: 00	DS0560W	(02)	13: 54: 00	ECSA % UTILIZATI ON(TOTAL) = 74 (>70) *****
13: 55: 01	DS0560W	(03)	13: 55: 00	ECSA % UTILIZATI ON(TOTAL) = 74 (>70) *****
13: 56: 00	DS0560W	(04)	13: 56: 00	ECSA % UTILIZATI ON(TOTAL) = 74 (>70) *****
13: 57: 01	DS0560W	(05)	13: 57: 00	ECSA % UTILIZATI ON(TOTAL) = 74 (>70) *****
13: 58: 00	DS0560W	(06)	13: 58: 00	ECSA % UTILIZATI ON(TOTAL) = 74 (>70) *****
13: 58: 12	DSN3201I		+ ABNORMAL EOT IN PROGRESS FOR USER=LGS11	
13: 58: 12	CONNECTI ON-ID=DB2CALL		CORRELATI ON-ID=LGS11	

Figure 37. Sample Log Display

You also can request an online audit trail of all *DB2 system messages* by specifying LOG=YES per target DB2 in the DMRBEX00 member of the BBPARM data set.

2. Check for exceptions that occurred in a specific period (Locate Time).

COMMAND ==> **T 1130**

3. FIND a specific problem (and press **PF5** for RFIND). For example:

COMMAND ==> **FIND DW0200W** (#SQLM monitor warning)

COMMAND ==> **FIND DSNT375** (DB2 deadlock messages)

COMMAND ==> **FIND DZ0630W** (Runaway TSO query)

COMMAND ==> **FIND RUNAWAY** (Runaway thread from any connection)

COMMAND ==> **FIND DWO** (All workload monitor warnings)

4. Scroll to the left with **PF10** to see the origin of the messages.

This value can be used to select a subset of the messages in the Journal log.

5. Type **PROFILE** on the COMMAND line to access the Enhanced Journal Facility, as shown in [Figure 38](#).

*Select  
Messages  
from  
One Target*

```

Boole and Babbage ----- Log Display ----- General services
COMMAND ===>
                                     Date --- 00/05/16
                                     Time --- 12:41:32

Included Origins                Excluded Origins

DB2G_____
_____
_____
_____
_____

                                     _____
                                     _____
                                     _____
                                     _____

                                     _____

Press END to SAVE Profile and return to application
CANCEL to discard changes
    
```

Figure 38. Enhanced Journal Facility

6. Type your **DB2 target name** (see the TGT field) in the Included Origins column. Return to the Log Display to view the messages only from that DB2.

7. Issue the **PROFILE** command again and specify **&TARGET** in the Included Origins column to automatically select the messages from the current target DB2 subsystem. Also specify the **BBI-SS PAS id** to include MAINVIEW for DB2 messages.



---

## Chapter 3. Monitoring a DB2 Data Sharing Group

Monitoring DB2 data sharing increases the complexity of both the environment and the number of tuning *knobs* that needs to be analyzed and optimized.

The first key requirement is to be able to focus on the level of data that is required to understand how the data sharing group is performing, or to solve a particular problem. Since a data sharing group consists of multiple DB2 members, and these members work together and share resources, viewing each DB2 individually is no longer adequate.

These scenarios show you how to look at all the members concurrently so you can easily compare activity and resource usage across the group, as well as access summarized data for the whole group, such as group buffer pool activity, total database I/O to the shared tables, or global lock contention. Of course, you will still drill down to an individual DB2 member for details as needed.

In this practice session, you

1. Define a Single System Image (SSI) context for the data sharing group(s).
2. Check on current group activity.
3. Look at page set considerations.
4. Analyze global lock contention.
5. Tune group buffer pools.

**Note:** This practice session takes approximately one hour to complete.

## Define the Group Context

Start this scenario at the EZDSSI menu, described on page 8.

The first thing you need to do is make sure that you have a Single System Image (SSI) context defined for the data sharing group(s) you want to look at:

1. Select the **Set SSI Context** hyperlink (first option under Tools And Menus).

This presents a view of all the defined SSI contexts for MVDB2, as shown in [Figure 40](#).

*All Defined  
MVDB2 SSI  
Contexts*

```

11MAY2000 16: 17: 22 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> CSR
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =CONASEL=====SYSD=====*=====11MAY2000==16: 17: 21====PLEXMGR=====4
CMD SSI      Product  Description                               Num  Num
--- Context- ----- Context- of Target--- -----
      ALL      MVDB2   DB2 SSI context                           3    3
      ALLDB2   MVDB2   All DB2 Systems                           9    9
      DBGHC    MVDB2   DB2 5.1 Data Sharing Group                 2    2
    
```

Figure 40. SSI Context Selection List (CONASEL)

2. If you do have a context defined for the data sharing group you want to monitor, all you need to do is hyperlink on the context name. This returns you to EZDSSI with the new context in effect.

If you don't have a context defined, you should define one now. (Even if you have a context defined, you may want to browse a little.) On the COMMAND line, type

**CONACTZ**

This view lists all SSI contexts (as known by your CAS and any connected CASs). Your CASID is shown on the window information line.

If you have multiple MAINVIEW products installed, here you see one of the most powerful features of SSI—that the same context can be defined for several different MAINVIEW products. Obviously, since each product may look at different target types, the selection criteria can vary.

3. Select the default context of **ALL** for **MVDB2** to hyperlink to the CONACT view.

Now you see each of the defined DB2 target subsystems, as shown in [Figure 41](#).

*All Defined  
DB2  
Subsystems*

```

11MAY2000 11: 05: 15 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> CSR
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =CONACTZ==CONACT==SYSD=====*=====11MAY2000==11: 02: 43====PLEXMGR=====3
CMD SSI      Product  Target  Status  Description
--- Context- ----- Context- of Target--- -----
      ALL      MVDB2   DBOHC   ACTIVE  MAINVIEW for DB2
      ALL      MVDB2   DB1HC   ACTIVE  MAINVIEW for DB2
      ALL      MVDB2   DB2HC   ACTIVE  MAINVIEW for DB2
    
```

Figure 41. SSI Context Activity Manager (CONACT)

4. Type **CONDEF** on the COMMAND line to access the context definition dialog.
5. Browse an example of a context definition.  
If a data sharing group context already exists, hyperlink on the SSI context name to see how the target filters were defined. If you don't see the group, select any other context. (You will at least see the default context of ALL.)
6. Press **PF3** to return to CONDEF.
7. Type **EDIT** on the COMMAND line to obtain an edit lock.
8. Type **ADD** on the COMMAND line to display the Add SSI Context Definition panel, as shown in [Figure 42](#).

### Add a New Context

```

11MAY2000  11: 10: 14  ----- INFORMATION DISPLAY -----
COMMAND  ===>                                     SCROLL  ===>  CSR
CURR WIN  ===>  1          ALT WIN  ===>
>W1  =CONDEF=====SYSD=====*(00 EDIT          )=====PLEXMGR=====10

----- ADD SSI CONTEXT DEFINITION -----
COMMAND  ===>

SSI Context  ===>
Description  ===>

Inclusion Filters: (Target is included if any are true)
  1TGTNAME = IN (DB1P, DB2P)
  2
  3
  4
  5
  6
  7
  8

Equivalent SUBSTITUTION parameters to be used in filter expression:
%1=TGTNAME      %2=TGTSYSTEM  %3=TGTPRODUCT  %4=TGTSERVER
%5=TGTDESC

Type  END to add the SSI context definition
      CANcel to leave without adding

```

Figure 42. Add SSI Context Definition Panel

In this panel, you define the context name and description and the selection filters for the targets that will be part of this context. Help is available on each of the fields.

You can define several filter conditions, but, in this case, all you will need is to filter by **TGTNAME** (the shorthand for this is **%1**). Since most sites use some kind of naming convention for the members, you probably only need one filter condition; for example:

**%1 = DB?P**

would include DB2s named DB1P, DB2P, and so forth.

If you prefer, you can define an IN list; for example:

**TGTNAME IN (DB1P, DB2P)**

would include only the DB2s named DB1P and DB2P.

## Define the Group Context

9. Type the following commands to activate this definition:

**END** (to return to the CONDEF view and complete the ADD)  
**SAVE** (to save the information)  
**INSTALL** (to dynamically activate this SSI context in this CAS)

10. If there are multiple CASs involved, you need to complete this definition in each CAS. (CASACT will show you a list of CASs and you can type **CON casid** to switch to another CAS.)

11. If they all share the same BBPARM (you still see the new definition after you switch to another CAS), you only need to type

**CONDEF**  
**EDIT**  
**INSTALL**

Otherwise, you must repeat the whole set of steps.

---

## Check Current Group Activity

Often you just want to check the health of each member of your data sharing group, and check on the activity levels. The previous dialogs covered how to check on multiple DB2s subsystems and their active threads. For data sharing, all you need to do is focus on just a data sharing group, instead of the default SSI context of ALL used earlier.

To view activity for a data sharing group:

1. You should have already set the context in the previous step. As a shortcut in the future, you can also simply type

**CON context** (example: CON DBGHC)

2. Now you can select options from the EZDSSI menu to look at just this group. Try the **Status Overview** option again.

## Look at Page Set Considerations (I/O / GBP-DEP)

Data sharing means that some or all of the DB2 tables can be accessed concurrently from all members of the group. But the tools provided by DB2 to understand the impact of this are limited. This section shows the power of what SSI can do to make this easier.

To look at page set considerations:

1. One of the key tuning areas in DB2 is I/O analysis. But looking at I/O for shared page sets or volumes from one DB2 member at a time does not give a complete picture.

Select **Volume I/Os (SSI)** from the data sharing section in EZDSSI to see a view of total I/O per volume from all members (PSVOLSSI), as shown in [Figure 43](#).

*Volume I/O  
for the Group*

```

11MAY2000 15:46:52 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =PSVOLSSI===== (DBGHC====*=====) 11MAY2000==15:46:37====MVDB2=====13

```

Volume	Sync I/0s	I/0 %	Sync Max I/0 Wait	Sync Avg I/0 Wait	0...20...40	Async I/0s	I/0 %	Async Pages
BAB309	6	0.4	31	17	*****	2	0.0	2
BAB310	309	19.9	354	23	*****	31	0.5	104
BAB311	3	0.2	38	20	*****	0	0.0	0
BAB312	1120	72.3	10322	32	*****	5732	91.8	44287
BAB314	8	0.5	98	29	*****	0	0.0	0
BAB317	36	2.3	176	27	*****	0	0.0	0
BAB318	5	0.3	27	16	*****	0	0.0	0
BAB322	12	0.8	38	14	***	449	7.2	3535
BAB325	9	0.6	52	16	*****	0	0.0	0
BAB330	8	0.5	104	25	*****	0	0.0	0
BAB331	8	0.5	30	15	***	19	0.3	143
BAB332	15	1.0	30	10	**	8	0.1	61
TSG314	11	0.7	31	16	*****	3	0.0	14

Figure 43. Volume I/O SSI Summary—Session (PSVOLSSI)

2. Select **one of the volumes** to see how many page sets are being accessed on that volume from each DB2 (PSVOLSZ), as shown in [Figure 44](#).

*Volume I/O  
per Member*

```

11MAY2000 15:50:40 ----- INFORMATION DISPLAY -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
>W1 =PSVOLSZ===== (DBGHC====*=====) 11MAY2000==15:50:39====MVDB2=====2

```

Volume	DB2	Nr. PSs	Sync I/0s	I/0 %	Sync Max I/0 Wait	Sync Avg I/0 Wait	0...20...40	Async I/0s
BAB312	DB1HC	13	69	4.1	10322	188	*****+	0
BAB312	DB2HC	17	1102	66.1	1796	22	*****	5732

Figure 44. Volume I/O Summary—Session (PSVOLSZ)

3. Select **Nr. PSs** for one of the members to see a list of each page set (PSVOLPS) with I/O data for just that member.
4. Press **PF3** to return to PSVOLSZ and select **the volume** instead.

Now you see a list of all page sets on that volume (PSVOLPSZ) and can see how many members access each. The I/O data here shows totals for the group.

5. Select **the volume** once again.

Now the list shows each page set with the I/O data broken down per member (PSVOLPS), as shown in [Figure 45](#).

*I/O per  
Page Set  
and Member*

```

11MAY2000 15:53:44 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =PSVOLPS===== (DBGHC====*=====) 11MAY2000==15:53:33====MVDB2=====30
-----Page Set----- DB2          Sync  Sync Max  Sync Avg
I/Os  I/O Wait  I/O Wait 0...20..40
BAB312 DSNDB01 DSNLLX02 001 DB1HC          4          29          12 *
BAB312 DSNDB01 DSNLLX02 001 DB2HC          11         28          24 **
BAB312 DSNDB01 DSNSTP01 001 DB1HC          5          28          17 **
BAB312 DSNDB01 DSNSTP01 001 DB2HC          8          29          15 **
BAB312 DSNDB01 SPT01      001 DB1HC          3          36          27 ***
BAB312 DSNDB01 SPT01      001 DB2HC          21         40          13 *
BAB312 DSNDB06 DSNADH01 001 DB1HC          3          43          17 **
BAB312 DSNDB06 DSNADH01 001 DB2HC          23         221         37 ****
BAB312 DSNDB06 DSNATX01 001 DB2HC          13         145         30 ***
BAB312 DSNDB06 DSNATX02 001 DB1HC          12         155         40 ****
BAB312 DSNDB06 DSNATX02 001 DB2HC          139         98          17 **
    
```

Figure 45. Volume Page Sets (PSVOLPS)

6. Press **PF3** to return to EZDSSI; then select the **Tuning Wizards** option, and then the **Data Sharing Wizard**.

7. There are several paths you could take on this panel, but first look down at the **Page Set Considerations** section. This provides another way to get to the Volume I/O data you just looked at, but we are now going to select the option to **Check GBP-DEP per Member** to see the level of sharing, as shown in [Figure 46](#).

*Check  
GBP-DEP  
per Member*

```

11MAY2000 16:01:31 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =WZDSDEP===== (DBGHC====*=====) 11MAY2000==16:01:30====MVDB2=====2
DB2      MVS          GBP  Local  Remote Log Recs  Ckpts To  Mins. To  Delete
Target  System          DEP R/W Int R/W Int  To Ckpt PCLOSE(N) PCLOSE(T)  Names
DB1HC   SYSC              4      4      4    50000      5      10      0
DB2HC   SYSC              4      4      3    50000      5      10      8
    
```

Figure 46. Data Sharing Wizard GBP-DEP/Member (WZDSDEP)

This shows a summary of how many GBP-dependent page sets there are in the group per DB2 member, and includes the key ZPARM values that affect how long a page set remains in this status.

- Tab the cursor to the column header, **Mins to PCLOSE(T)**, and press **PF1** to get field help on this value, as shown in [Figure 47](#).

Many field help panels, like this one, contain tuning tips in addition to the field definition.

*Tuning Tips*

```

11MAY2000 16: 01: 31 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> CSR
CURR WIN ==>> 1          ALT WIN ==>>
W1 =WZDSDEP===== (DBGHC====*=====) 11MAY2000==16: 01: 30====MVDB2=====2
DB2      MVS          GBP   Local  Remote Log Recs  Ckpts To  Mins. To  Delete
Target   System      DEP R/W Int R/W Int  To Ckpt PCLOSE(N) PCLOSE(T)  Names
DB1HC    SYSC
DB2HC    SYSC          Help      Pseudo-close Timer Parameter      Help
Command ==>>                                Scroll ==>> CSR
-----

This value lists the amount of time, in minutes,
that must elapse before a data set can be a
candidate for pseudo-close (QWP1TMR).

ZPARAM name: PCLOSET in DSN6SPRM

Tuning Tip: The PCLOSEN and PCLOSET parameters are
associated with pseudo-close. PCLOSEN
defaults to 5 system checkpoints and
PCLOSET defaults to 10 minutes. There
is a trade off in setting these
parameters. If these values are set
too high, data sets may remain
GBP-dependent for longer than
necessary and incur unnecessary
performance overhead. If these values
    
```

Figure 47. Field Help

- Press **PF3** to return to the Data Sharing Wizard (WZDSHAR) and select the option to **Check GBP-DEP by GBP/PS**.

*Per Group  
Buffer Pool*

```

11MAY2000 16: 18: 16 ----- INFORMATION DISPLAY -----
COMMAND ==>>                                SCROLL ==>> CSR
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =PSBPGBPZ===== (DBGHC====*=====) 11MAY2000==16: 18: 15====MVDB2=====2
Bfrpl  DB2      Nr.   GBP      Local      Remote      VP      VP
ID     Target   PSs  Dependent R/W Interest R/W Interest Current Changed
BPO0  DB1HC     56   2         1           1         1     200    0
BPO0  DB2HC     72   2         1           1         1     88     1
    
```

Figure 48. Buffer Pool Page Set GBP-DEP Summary (PSBPGBPZ)

This summarizes the same data, also for the whole group, but now organizes it per group buffer pool and member.

- Select one pool (**Bfrpl ID**) to see a list of all the open page sets in that pool for all targets.

- Press **PF3** to return to PSBPGBPZ and then hyperlink on the **GBP Dependent** column to list only the GBP-dependent page sets in that pool (PSGBP), as shown in [Figure 49](#).

*Open  
Page Sets  
in One Pool*

```

11MAY2000 16:19:22 ----- INFORMATION DISPLAY -----
COMMAND ==>>
CURR WIN ==>> 1          ALT WIN ==>>
>W1 =PSBPGBPZ=PSGBP===(DBGHC====*====) 11MAY2000==16:18:15====MVDB2=====4
-----Page Set----- DB2          No. GBP Local  Remote  VP      VP
Database Object  Prt Target  Ty Usrs Dep Interest Interest Current Changed
DSN8D51A DSN8S51E 001 DB1HC  PS   0 Y R/O   R/W      1       0
DSN8D51A DSN8S51E 001 DB2HC  PS   0 Y R/W   R/O      0       0
DSN8D51A DSN8S51P 001 DB1HC  TS   0 Y R/W   R/O      2       0
DSN8D51A DSN8S51P 001 DB2HC  TS   0 Y R/O   R/W      0       0
    
```

Figure 49. Page Set GBP-DEP Status (PSGBP)

You can not only see which member(s) is GBP-dependent, but also which member(s) has Read/Write Interest and how many pages from that page set are cached in each member's local buffer pool.

## Analyze Global Lock Contention

One of the most important issues in data sharing is to control global lock contention, since it not only causes the usual problems of application availability you deal with in one DB2, but can significantly degrade performance in a data sharing group as well.

To analyze global lock contention:

1. Press **PF3** to return to the Data Sharing Wizard (WZDSHAR) and select **Lock Contention Wizard**.

The Global Lock Wizard (WZLKGOPT) is displayed, as shown in [Figure 50](#).

*Analyze  
Contention  
for a Group*

```

11MAY2000 16:20:50 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1      ALT WIN ==>
W1 =WZLKGOPT===== (DBGHC=====*)===== 11MAY2000==16:20:49====MADB2=====1
** LOCK WIZARD - GROUP **      Target Values... DB1HC

. Check Member Contention?      Interval      Sessi on
  (& Select any Member)          Global Locks... 243      12900

. Review History and set TIME?    Global Suspends. 22      1025
  (If Current Interval OK)

. Any Gross Locks? (2 hrs)        GrossLocks..... 0      0
  (Reduced Concurrency)

. List Group Lockouts?           Timeouts..... 0      1
  (Contention Victims)           Deadlocks..... 0      3

. Which Resources Involved?
  (Data Hot Spots)

. Compatible Workloads?
  (Connections Involved)

Analyze Plans In Lockouts?
. Global Blockers/Waiters
. Blocker Plans
. Waiter Plans
    
```

Figure 50. Global Lock Wizard (WZLKGOPT)

The Global Lock Wizard helps you step through the analysis of lock contention in DB2 for a data sharing group. It allows you to look at current status or choose an earlier time interval, perhaps one that a user of your system has complained about.

A history of the most important symptoms of lock problems—timeouts and deadlocks—is available, as well as information on system considerations and statistics that can point out potential causes of problems.

2. Select **Check Member Contention?**

This view, shown in [Figure 51](#), enables you to see the key indicators for all of the members together, allowing you to identify quickly whether or not there are any global contention issues to analyze. The counts shown are for the time since DB2 started.

**Check Member Contention**

```

11MAY2000 16:21:53 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =WZLKGOPT=WZLKGMEM(DBGHC====*=====) 11MAY2000==16:20:49====MVDB2=====2
DB2      MVS      Total   Local   Global   % of Group  Global False
Target  System  Lockouts Suspends Suspends . . . . 0 . . . 50 . . 100 Locks Cont.
DB1HC   SYSC           4       42      1025   53.9 ***** 12900 437
DB2HC   SYSC           6       85       877   46.1 ***** 27809 348
    
```

Figure 51. Lock Wizard Global Members (WZLKGMEM)

3. From here, you can hyperlink on any one DB2 to analyze its contribution to (or victimization by) global lock contention, as shown in [Figure 52](#).

**Member Lock Detail**

```

11MAY2000 16:23:49 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =WZLKGOPT=WZLKGCON(DBGHC====DB2HC===) 11MAY2000==16:20:49====MVDB2=====1
Global / Local Contention
Interval          Sessi on
Lockouts
Timeouts          0              3
Deadlocks         0              3
Indicators
% Global Contention 5.6            1.5
. If High - Tune GBP-DEP
% False Contention 23.3           39.7
. If High-Tune CF (MMVS)
Global Lock Activity
XES Sync Requests 239            27809
XES Async Requests 0              7
IRLM Suspensions 23             513
XES Suspensions  0              16
False Suspensions 7              348
. More Info...
Local Lock Activity
Lock Requests     21959          121627
Lock Suspensions 0              4
Latch Suspensions 10             57
. More Info...
    
```

Figure 52. Lock Wizard Global/Local Contention (WZLKGCON)

This view provides the information to analyze both global and local lock contention for a selected member. Besides the most critical statistics, both for the current interval and since DB2 startup, the two key indicators of % global contention and % false contention are calculated for you. Thresholds are defined to highlight any value that exceeds the recommended value. Place the cursor on the **% Global Contention** header or fields to see an explanation of the calculation and what it means.

## Analyze Global Lock Contention

- Press **PF3** to return to the Lock Wizard - Group panel (WZLKGOPT) and select **List Group Lockouts?** to see a list of the latest timeouts and deadlocks that have occurred in all the members of the group.

The *Global Contention* flag on the right is set to YES when the conflict occurred between threads running on different members, as shown in [Figure 53](#).

### Group Lockouts Showing Global Contention

```

11MAY2000 16: 25: 20 ----- INFORMATION DISPLAY -----
COMMAND ===>                                SCROLL ===> CSR
CURR WIN ===> 1          ALT WIN ===>
>W1 =LKEVSSI===== (DBGHC====*=====) 11MAY2000==16: 25: 20====MVDB2=====7
Date / Time      Lockout  Victim  Victim  Victim  Victim  Victim  Global
----- Type      Member  Plan    AuthID  Connect CorrName Cont.
11MAY-16: 12: 37 DEADLOCK  DB2H    RXDB2   BOLLAA2 DB2CALL DMRDLK1 YES
11MAY-16: 10: 18 TIMEOUT  DB1H    RXDB2   BOLLAA2 DB2CALL DMRTMO2 YES
11MAY-16: 10: 15 TIMEOUT  DB2H    RXDB2   BOLLAA2 DB2CALL DMRTMO3
11MAY-16: 06: 43 TIMEOUT  DB2H    RXDB2   BOLLAA2 DB2CALL DMRDLK3
11MAY-16: 06: 42 DEADLOCK  DB2H    RXDB2   BOLLAA2 DB2CALL DMRDLK1 YES
11MAY-15: 45: 46 DEADLOCK  DB2H    RXDB2   BOLLAA2 DB2CALL DMRDLK1 YES
11MAY-15: 45: 26 TIMEOUT  DB2H    RXDB2   BOLLAA2 DB2CALL DMRDLK3

```

Figure 53. Global Lockout Events (LKEVSSI)

- Press **PF3** to return to the Lock Wizard - Group panel (WZLKGOPT) and select **Which Resources Involved?** to analyze resource conflicts in the group, as shown in [Figure 54](#).

### Group Resource Conflicts

```

11MAY2000 16: 27: 13 ----- INFORMATION DISPLAY -----
COMMAND ===>                                SCROLL ===> CSR
CURR WIN ===> 1          ALT WIN ===>
>W1 =LKRESZ===== (DBGHC====*=====) 11MAY2000==16: 27: 12====MVDB2=====2
--Resource Name-- Total      % Total      Global
Database Object  Conflicts  0.. 50.. 100 PAGE  ROW INDEX Conflicts
DSN8D51A DSN8S51E      8  72.7  *****      8    0    0      5
DSN8D51A DSN8S51P      3  27.3  ***           0    3    0      3

```

Figure 54. Lockout Resource Summary (LKRESZ)

The first view summarizes all conflicts by table space / index space, so that you can quickly identify the objects with the most contention. The count on the right shows how many conflicts exist between members (global).

- Hyperlink on a **resource name** showing several conflicts to see a breakdown of these conflicts by specific resource, down to a page or row level (LKRESNRZ).

With this view, hot spots in your tables are immediately visible, as shown in [Figure 55](#).

### See Hot Spots

```

11MAY2000 16: 28: 44 ----- INFORMATION DISPLAY -----
COMMAND ===>                                SCROLL ===> CSR
CURR WIN ===> 1          ALT WIN ===>
>W1 =LKRESZ==LKRESNRZ(DBGHC====*=====) 11MAY2000==16: 27: 12====MVDB2=====1
--Resource Name-- Resource  Resource  Total      % Total      Global
Database Object  Number  Type      Conflicts  0.. 50.. 100 Conflicts
DSN8D51A DSN8S51E 000001200 DATAPAGE      8  72.7  *****      5

```

Figure 55. Lockout Resource Number Summary (LKRESNRZ)

7. Hyperlink on either the **resource name** or the **resource number** to view a list of each lockout event that involved this resource (LKRESD), as shown in [Figure 56](#).

*Each Event for  
This Resource*

```

11MAY2000 16:30:21 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> CSR
CURR WIN ==> 1           ALT WIN ==>
>W1 =LKRESZ===LKRESD==(DBGHC====*====) 11MAY2000==16:27:12====MVDB2=====8
--Resource Name-- Resource  Resource  Time  Lockout  Blocker  Waiter  Gbl
Database Object  Number  Type    -----  Type    PlanName PlanName Con
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16:12:37 DEADLOCK RXDB2   RXDB2   Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16:10:18 TIMEOUT  RXDB2   RXDB2   Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16:10:18 TIMEOUT  RXDB2   RXDB2   Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16:10:15 TIMEOUT  RXDB2   RXDB2
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16:06:43 TIMEOUT  RXDB2   RXDB2
DSN8D51A DSN8S51E 0000001200 DATAPAGE 16:06:42 DEADLOCK RXDB2   RXDB2   Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 15:45:46 DEADLOCK RXDB2   RXDB2   Yes
DSN8D51A DSN8S51E 0000001200 DATAPAGE 15:45:26 TIMEOUT  RXDB2   RXDB2

```

Figure 56. Lockout Resource Conflict Detail (LKRESD)

This view helps you quickly determine which plans are involved in the contention and whether or not the problem was occurring only at a particular time, perhaps because of an application affinity problem.

8. Press **PF3** to return to the Lock Wizard - Group panel (WZLKGOPT) and select **Global Blockers/Waiters?** to see which plans are involved in the lockouts, as shown in [Figure 57](#).

*Which Plans  
Involved?*

```

11MAY2000 16:31:40 ----- INFORMATION DISPLAY -----
COMMAND ==>                                     SCROLL ==> CSR
CURR WIN ==> 1           ALT WIN ==>
>W1 =LKBWZSSI===== (DBGHC====*====) 11MAY2000==16:31:37====MVDB2=====3
Blocker Blocker Waiter Waiter  Timeout Deadlock  % Conflicts  Gbl
PlanName Member  PlanName Member  Invol v.  Invol v.  . . . . 0 . . . 50 . 100  Con
RXDB2   DB1H   RXDB2   DB2H           1         3   36.4  ****         4
RXDB2   DB2H   RXDB2   DB1H           1         3   36.4  ****         4
RXDB2   DB2H   RXDB2   DB2H           3         0   27.3  ***          0

```

Figure 57. Lockout Global Blocker Waiter Summary (LKBWZSSI)

This view lists a summary of blocker/waiter plans and systems. You can identify conflicting plans across multiple data sharing DB2 members. Again, the list of events for a particular combination is available with a hyperlink, so you can quickly check to see if there are any application scheduling problems.

## Tune Group Buffer Pools

After global locking, the next most important area of data sharing tuning is to determine whether or not the group buffer pools are the right size and have the correct ratio of directory to data entries to support your workload, based on the amount of inter-DB2 sharing that is occurring.

If any of these resources is lacking, overhead increases in the group buffer pools, the coupling facility, and the local pools. It can also cause unnecessary I/O.

To tune the group buffer pools:

1. Press **PF3** to return to the Data Sharing Wizard (WZDSHAR) and select **Review Group (Session)** to see an overview of key statistics about
  - Global Contention
  - Group buffer pool performance
  - GBP-dependent page sets
  - Coupling facility activity (castouts, failures)

*Data Sharing  
Member  
Overview*

```

11MAY2000 16:33:40 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =WZDSGMEM===== (DBGHC====*=====) 11MAY2000==16:33:40====MVDB2=====2
DB2      MVS      Global  Total      Read Hit %  GBP
Target   System   Cont. %  Lockouts  . . . . . 0 . . 50 . . 100  DEP Castouts Failures
DB1HC    SYSC      3.76    4         14.0 **          1         4         0
DB2HC    SYSC      1.57    6         22.2 **          1         23        0
    
```

Figure 58. Data Sharing Wizard Group Members (WZDSGMEM)

This view shows you some of the key indicators per member. This allows you to identify quickly whether or not there are any performance issues to analyze. The counts shown are for the time since DB2 was started.

From here, you can hyperlink on any one DB2 to see all the member statistics for both the current interval and the session data since DB2 startup. Or you can return to the first panel to follow some of the analysis paths.

- Press **PF3** to return to the Data Sharing Wizard (WZDSHAR) and select **Check GBPs** to access the GBP analysis section of the Data Sharing Wizard.

This decision panel provides options to analyze GBP size and the ratio of directory to data entries, as shown in [Figure 59](#).

### GBP Analysis

```

11MAY2000 16:35:20 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1      ALT WIN ==>
WI =WZDSBOPT===== (DBGHC====DB***** ) 11MAY2000==16:35:19====MVDB2=====1
** DATA SHARING WIZARD - GBPs      Group GBP Values

Check Totals per GBP
  (Select GBP for Detail)

          Interval      Sessi on
. GBP Size Too Small?      Read Hit % . . . . . 0.0      6.2
  (Low Hit% / Cache Full)  Miss-Cache Full.    0         0
                          Curr. Chngd Pgs.    0
. Too Few Directory Entries?  Dir. Reclai ms. . . . 0         0
  (Reclai ms Cause XI/DASD Reads)  XI s/Dir. Recl. . . . 0         0
. Too Few Data Entries?      Castouts. . . . . 6         95
  (Castouts More Frequent)       Write Fail-Stor.    0         0
  (Castout Too Slow)

```

Figure 59. Data Sharing Wizard GBP Options (WZDSBOPT)

There are three diagnostic paths here, but, as an example, we are going to follow the path to analyze directory entries. Even if the pools are large enough, a lack of directory entries (used to register each page) can cause problems.

The key indicators on the right are there to help you decide whether or not you need to do additional analysis by following one or more paths. On this panel, all of these values are for the whole group, summarized for all GBPs.

Possible symptoms of too few directory entries are

- The occurrence of directory reclaims so that new pages can be registered
- The even worse consequence—that cross-invalidations of pages in the members' local buffer pools are occurring because of these directory reclaims

If one of these indicators is greater than 0, the field is highlighted in red.

- Hyperlink on **Too Few Directory Entries** to see a tabular list of all defined group buffer pools, as shown in [Figure 60](#).

**GBP Group Overview**

```

11MAY2000 16:36:09 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
>W1 =WZDSGBR===== (DBGHC====*=====) 11MAY2000==16:36:09====MVDB2=====9
  GBP      Curr.  Pend.   Dir.   Data  Directory  XI from  XI Read  Read Hit
  Name     Ratio  Ratio  Entries  Entries  Recl ams  Dir. Recl.  Miss    Ratio
BP0         5      5      942     187         0         0         3      6.2
BP1         0      0         0         0         0         0         0      0.0
BP2         0      0         0         0         0         0         0      0.0
BP4         0      0         0         0         0         0         0      0.0
BP5         0      0         0         0         0         0         0      0.0
BP9         0      0         0         0         0         0         0      0.0
BP11        0      0         0         0         0         0         0      0.0
BP32K       0      0         0         0         0         0         0      0.0
BP32K9      0      0         0         0         0         0         0      0.0
    
```

Figure 60. Data Sharing Wizard GBP Directory Entries (WZDSGBR)

Now you can see each group buffer pool with the current definitions, the two key indicators we saw before for all pools (Directory Reclaims and XI from Dir.Recl. columns), and some additional related statistics.

- From here you can select a single GBP for further analysis, as shown in [Figure 61](#).

**GBP Directory Entries (Group)**

```

11MAY2000 16:37:11 ----- INFORMATION DISPLAY -----
COMMAND ==>
CURR WIN ==> 1          ALT WIN ==>
W1 =WZDSGBRD===== (DBGHC====DB*****) 11MAY2000==16:37:11====MVDB2=====1
  GBP Directory Entry Analysis - Detail  BP0

                                Interval          Sessi on
Recl ams for Directory Entries?         0              0
  (GBP Page/Dir Reused)

Directory Recl ams Causing XI?         0              0
  (Local VP Pages Invalidated)

Also Increasing DASD Reads?
. Check Members for:
  * XI Miss ==> DASD Reads

GBP Hit Ratio Low?
. Check Members for:
  * Low GBP Hit Ratio

Definitions
Total Size (4K Blocks)          256
Directory Entries                942
Data Entries                     187
Current Ratio                    5
Pending Ratio                    5
    
```

Figure 61. Data Sharing Wizard GBP Directory Detail (WZDSGBRD)

This analysis panel provides details to help you determine whether or not a group buffer pool may have too few directory entries to support its part of the data sharing workload. It addresses the main issue—whether directory reclaims are causing cross-invalidations (XI) and probably increasing I/O.

In this panel, you now see the two key indicators again, but both as interval (current activity) and session (total since DB2 startup) counts.

The answer to the additional question of whether or not this is increasing DASD reads can only be seen by looking at statistics per member instead of at the group level.

5. Hyperlink on **Check Members for XI Miss** to see data sharing session activity counts per member for this group buffer pool, as shown in [Figure 62](#).

*Counts per  
GBP and  
Member*

```

11MAY2000 16:39:04 ----- I N F O R M A T I O N   D I S P L A Y -----
COMMAND ===>
CURR WIN ===> 1          ALT WIN ===>
W1 =WZDSGBSA===== (DBGHC====*=====) 11MAY2000==16:39:02====MVDB2=====2
GBP  DB2      Read  XI Read  NF Read  Changed  Clean Sync Read Async Read
Name Target  Hit %   Miss   Miss    Writes  Writes   Total   Total
BPO  DB1HC    14.0    3      40      12       0       50      0
BPO  DB2HC    22.2    1      13      20       0       18      0

```

Figure 62. Data Sharing Wizard GBP Size Activity (WZDSGBSA)

The XI Read Miss column now shows whether or not any members had to do additional I/O to read in pages lost through reclaims and cross-invalidation. Non-zero counts are highlighted. From here, you can also hyperlink on a GBP to see interval and session counts for that GBP.

You have seen a few of the most important areas for data sharing monitoring. You have used a Single System Image context to enable you to monitor all members of a group, both individually and summarized, and to pull all the relevant data together. Now you have the tools to master the new and complex data sharing performance issues.



---

## Chapter 4. Tuning an Application with Trace

These scenarios teach you how to navigate easily through the MAINVIEW for DB2 trace setup panels and the displays of trace data, and acquaint you with the trace print utility.

In this practice session, you

1. Start an application trace and review the available options.
2. Review tips on running your tests with trace.
3. Look for application problems using the various trace displays.
4. Print a trace report for offline review.

**Note:** To complete the first exercise, DB2TRACE=D or ALL must be specified in your BBPARM USERID member. DB2TRACE=ALL must be specified to perform the I/O trace in a later exercise.

You must have authorization to update your BBPARM data set and issue the RESET USERID command (.E A userid) to make these changes. Contact your system administrator if you do not have authorization.

This practice session takes approximately two hours to complete.

## Start an Application Trace

To start an application trace, begin by checking the current traces:

1. From the Primary Option Menu, select the **TRACES** option.

OPTION ==> **4**

This panel lists all the current traces, as shown in [Figure 63](#). They may be active and collecting data from DB2 (STATus is ACTV), or already complete but not yet purged (STATus is COMP). We will return here later to view your own trace.

*Current Traces*

*ST Command*

BOOLE AND BABBAGE	-----	CURRENT TRACES	-----	PERFORMANCE MGMT
COMMAND ==>				TGT ==> DB2G
		INPUT	INTVL ==> 3	TIME -- 14:31:36
COMMANDS:	<b>ST (START APPLICATION TRACE)</b> ,	HT (HISTORY TRACES),	TYPE	
LC CMDS:	S (SELECT), W (SHOW),	M (MODIFY), I (SWITCH),	Z (STOP)	
	P (PURGE), R (REPLICATE),	H (HELP), Q (QUIESCE LOGGING)		
LC	PARM	TITLE	USER ID	TARGET TYPE AREA STAT LOG
	SUMMARY	DAILY ACCOUNTING TRACE	CI R11	DB2D SUM WKLD ACTV ACT
	DETAIL	TRACE OF APDT001T	PWW1	DB2D DET WKLD ACTV
	BIGELAP	ELAPSED OVER 10 SEC	PWW1	DB2D SUM WKLD ACTV

Figure 63. Current Traces Application

2. Transfer to **START APPLICATION TRACE**.

COMMAND ==> **ST**

You must be authorized to start a trace and to specify certain options. See your system administrator for the trace privileges you have. If you cannot start a trace, just browse these instructions and then go to [“Look for Application Problems” on page 69](#) to view an already active trace.

## Specify Options

On the Start DB2 Trace Request panel, shown in [Figure 64](#), you can specify various options for your trace. Many traces can be started using just this first panel. If you want to review additional options, they are available on three more panels that are accessed much like DB2I SPUFI options.

*Start Trace*

*Using Defaults*

```

BOOLE AND BABBAGE ----- START DB2 TRACE REQUEST ----- PERFORMANCE MGMT
COMMAND ==>>>                                                    TGT==> DB2G

PARM   ==>                (Trace identifier)      START ==>                (hh: mm: ss)
TYPE   ==> S              (S-Summary, D-Detail)   STOP  ==>                (hh: mm: ss/#mi n)
STORAGE ==> 1000K        (Display buffer size)   WRAP  ==> YES           (Y/N wrap buffer)
LOGTRAC ==> N          (Y/N log trace)          RST   ==> HOT           (HOT, PUR, QIS)
TITLE  ==> DB2 APPLICATION TRACE

Specify Selection Criteria:
DB2PLAN ==>
DB2AUTH ==>
DB2CONN ==>
DB2CORR ==>
DB2LOC  ==>
DB2PKG  ==>
CONNTYPE ==>

Specify additional trace options:      (* = processed)
Exception Filters                      ==> N (Y/N)
Detail Trace Options                   ==> N (Y/N)
Trace Log Data Set Options             ==> N (Y/N)

Press ENTER to process; END to cancel

```

Figure 64. Start DB2 Trace Request Panel

## Start an Application Trace

### 1. Define a trace using mostly defaults.

#### a. PARM ==> **id**

You can specify any name as an ID, but try to make it descriptive. For example, use your initials or an acronym for the application being tested, plus a number to identify the test: JNL003, ABCTEST1, ABC2IO.

#### b. START ==>

Leave blank to start the trace immediately.

#### c. TYPE ==> **D**

A Detail trace collects the DB2 accounting record and additional events (DB2 performance trace IFCIDs) per thread. The default includes the basic start and end events, plan allocation data, exceptions (like timeouts), sorts, and all the SQL statements. This can be modified on a later panel.

#### d. STOP ==> **10**

This traces for 10 minutes and then stops collecting data. Even if you decide not to log a trace, the data is available for display until the trace is purged.

#### e. STORAGE ==> **1000K**

Use the site default size for the storage buffer (used for online display).

#### f. WRAP ==> **Y**

Use the default to wrap the data in the STORAGE buffer if it fills up.

#### g. LOGTRAC ==> **Y**

Specify Y to request trace logging. This allocates a VSAM log for this trace. You can recall the data for online display or print reports until you decide to delete the data set.

#### h. RST ==> **HOT**

Use the default to restart the trace automatically without loss of data if DB2 goes down and up while you are tracing.

#### i. TITLE ==> **user-specified title**

This field is filled in with the default, but you should specify a title that will help you later to identify the contents of this trace.

#### j. DB2AUTH ==> **userid, SYSOPR**

Specify your user ID to trace your own tests. Add SYSOPR to trace prefetch reads. You can also specify other selection criteria. For a detail trace, you must specify DB2PLAN or DB2AUTH to limit DB2 tracing. This is subject to DB2 restrictions (1 plan / 8 authids, or vice versa).

**Note:** If you qualify by plan, prefetch read I/O events cannot be captured.

### 2. Request the additional trace options to review the other panels.

Exception Filters ==> **Y**

Detail Trace Options ==> **Y**

Trace Log Data Set Options ==> **Y**

### 3. Press **Enter** to view the next panel.

4. Review the Exception Filters panel, shown in [Figure 65](#), but leave it empty.

*Narrowing  
the Trace*

```

BOOLE AND BABBAGE ----- DB2 TRACE EXCEPTION FILTERS ----- PERFORMANCE MGMT
COMMAND ==>>>                                                    TGT -- DB2G

Specify Exception Filters:
ELAP ==>                GETPAGE ==>                MAXLOCK ==>                SQLDDL ==>
CPU ==>                 PGUPD ==>                 LOCKTBL ==>                SQLDYN ==>
ABORT ==>              READIO ==>                 INCRBIND ==>              SQLCTL ==>
UIDCOM ==>            LOCKSUSP ==>              SQLSEL ==>                SQLFETCH ==>
GETRIO ==>            LOCKESCL ==>              SQLUID ==>                SQTOT ==>
RLF ==>               TIMEOUT ==>              RIDFAIL ==>              CLAIMDR ==>
PWAITIO ==>          PWAITLK ==>              PWAITPF ==>              PWAITOT ==>
PFREQS ==>          PFREADS ==>              HPFAILS ==>              PRLGRP ==>
PRLRED ==>          PRLFALB ==>

```

Figure 65. DB2 Trace Exception Filters Panel

Specifying filters causes thread accounting records that do not meet the qualification to be discarded. The value can be a maximum (n) or a minimum (<n). For example, specifying ELAP ==> 10 only keeps threads with an elapsed time greater than 10 seconds. This is very useful when you are looking for poorly performing applications in an existing DB2 workload.

**Note:** If you specify several filters, they are ORed. A trace record is retained if any one of the comparisons is valid.

5. Press **Enter** to view the next panel.

6. Review the Detail Trace Options panel, shown in [Figure 66](#).

*Adding Detail Events*

```

BOOLE AND BABBAGE ----- DETAIL TRACE OPTIONS ----- PERFORMANCE MGMT
COMMAND ==>>                                     TGT -- DB2G

Specify additional events:
SQL           ==>> Y (Y/N)
SCANS        ==>> N (Y/N)
I/O          ==>> N (Y/N)
LOCKS        ==>> N (Y/N)
DDF          ==>> N (Y/N)
DDFVTAM      ==>> N (Y/N)

Specify event compression:
GROUP SQL     ==>> Y (Y/N)

Specify data collection buffer options:
TRSIZE       ==>> 400K (Trace buffer size)
TRBUFF       ==>> 20  (# of trace buffers)
    
```

Figure 66. Detail Trace Options Panel

TRSIZE ==> (value shown is the default for your site)

This is the size of a data collection buffer. Without logging, the data collected for one transaction (thread) is limited to two buffers. If you are tracing long-running applications, you may need to increase the size. With logging, multiple buffers can be written per thread and combined automatically when recalled online or printed.

TRBUFF ==>

You may need to increase this value to trace more concurrent threads or to provide multiple buffers for the logging of long-running applications. The recommended number is three or more times the number of concurrent threads to be traced.

GROUP SQL ==> Y

Y groups many consecutive identical SQL statements together to save space and make the event trace easier to read; for example, combine many FETCHes.

7. Specify the additional DB2 events you want to trace.

```

SQL           ==> Y      Default is Y for standard application tuning
SCANS        ==> Y      Default is N, specify Y this time
I/O          ==> Y      Default is N, specify Y this time
LOCKS        ==> N      Default is N, leave as is - very expensive
DDF          ==> N      Default is N, needed only for distributed work
DDFVTAM      ==> N      Default is N, needed only for DDF VTAM analysis
    
```

Each of the other groups of events adds additional overhead. You can specify any combination; for example, SQL and I/O but no SCANS.

8. Press **Enter** to view the next panel.

9. Review the Trace Log Data Set Options panel, shown in [Figure 67](#), but leave the defaults.

*Logging  
the Trace*

```

BOOLE AND BABBAGE ----- TRACE LOG DATA SET OPTIONS ----- PERFORMANCE MGMT
COMMAND ==>                                                    TGT -- DB2G

Number of Logs ==> 1          (# data sets; >1 for auto switch when full)
First Log DSN ==>
                        Low level qualifier of DSN must be V01
                        Blank for default: CIR7.DB1D.BLANK.mmmdd.Thhmm.V01
                        Names without quotes will be prefixed with CIR7

Overwrite logs ==> Y      (Y/N) (Action when all logs used)
Archive PROC ==>          (Blank for none/PROCLIB member name)
Log switch time ==>      (HH:MM that a log switch is requested)

Disposition ==> NEW (OLD/NEW) If NEW, specify options below:

Volumes ==> (V00001, V00002, V00003)
Primary CYLS ==> 5          SMS Storage Class ==> SMSSTOR
Data DSN Suffix ==> D1      SMS Data Class ==> SMSDATA
                               SMS Management Class ==> SMSMGMT

```

Figure 67. Trace Log Data Set Options Panel

a. Number of Logs ==> **1**

Multiple logs are usually needed only for continuous system traces for workload history.

b. First Log DSN ==>

Leave this blank to take the generated default. You must be authorized for dynamic allocation of a trace log. See your system administrator. If you are not authorized, there is a batch job (JXT011) to preallocate a log data set. Then type the name here and change DISPOSITION to OLD.

c. Overwrite Logs ==> **Y**

Y allows the latest trace data to be kept if it doesn't all fit in the log. N keeps the earliest trace data by quiescing the trace when full.

d. Archive PROC ==>

Leave blank. Not needed for simple single-log traces.

e. Log Switch Time ==>

Leave blank. Not needed for simple single-log traces.

f. Disposition ==> **NEW**

Leave NEW unless you had to preallocate a log.

g. Volumes ==>

This should be filled in with the default volume(s) for your site. If not, specify a volume the MAINVIEW for DB2 product address space (BBI-SS PAS) is allowed to use.

h. Primary Cyls ==>

Use the default.

## Start an Application Trace

10. Specify SMS values if necessary in your shop and defaults are not set.

```
SMS Storage Class    ==>
SMS Data Class       ==>
SMS Management Class ==>
```

11. Press **END** to process the options and return to the first panel.

## Activate the Trace

All options are now specified for your trace. The trace options are set to \*. If you want to drop the related options you specified, change the \* to **N**. To view them again, change the \* to **Y**.

When all options are \* or **N**:

1. Press **Enter** to submit the trace request.
2. Press **PF3** to return to the CURRENT TRACES panel.
3. Press **Enter** until the status of your trace changes to **ACTV**.

Data can now be collected. Enter some activity to be traced.

- If the status changes to **INV** (invalid), there was an error in your request that was not found in the preliminary syntax checking. An example of this kind of error would be a trace log data set allocation error because no space was available on the specified volume. You can see the detailed error messages on the Journal Log (press **PF5** to view).
- If the status changes to **QIS** (quiesced), the target DB2 is not up.

---

## Review Tips on Tracing Tests

There are many different scenarios for testing. Here are some hints on tracing a few of the most common tests.

- Testing your new application during development
  1. Qualify the trace by your own AUTHID. DB2 trace overhead is only incurred for the tests of your application. Depending on the number of users of the MAINVIEW for DB2 trace in your test DB2 system, you may be able to keep your trace request active through several test iterations, consolidating them on one trace log. (Only four detail traces can be active at one time.)
  2. BIND your program after the trace is started. This captures the text and EXPLAIN data of all static SQL statements in your trace output for reference when analyzing performance. EXPLAIN=YES is not required.
  3. If you don't set an automatic stop time when requesting your trace, don't forget to stop it when you have completed testing (use the Z line command in the CURRENT TRACES option). If you have logged the trace data, you should also purge the trace when complete. You can still browse and print the data from the log (HISTORY TRACES option). Otherwise, analyze the data from the online buffers and purge the trace when you are done.
- Analyzing the performance of an existing application
  1. Qualify the trace by PLAN.
  2. Set automatic start and stop times to cover a time span in which you expect sufficient activity.
  3. If the application usually runs well with only a few occurrences of poor performance, specify exception filters to save just those that you need to analyze. For example, select only those with high elapsed or CPU times, or with many GETPAGE requests or I/Os.
- Improving the performance of a bad SQL statement
  1. You may have identified a poorly performing SQL statement and now want to try out several different variations in the syntax to determine which is best. If you have RxD2/FlexTools or a similar tool, you can modify and execute the statement directly from the source while in ISPF edit. Otherwise, make the statement executable from SPUFI.
  2. Start the trace qualified by your AUTHID.
  3. Modify and execute each variation as dynamic SQL. You want to force the thread to terminate so an accounting record will be produced. This is done automatically with RxD2. With SPUFI, you must exit each time to terminate the thread.
  4. The trace automatically captures the SQL text, EXPLAIN data (with cost factor), and the accounting record and SQL performance statistics.

You now have all variations captured together in one trace for easy comparison.

## Review Tips on Tracing Tests

- Comparing test to production
  1. Save the detail trace log from the last test run. Run a short detail trace in production after cutover to compare performance. You can either print batch reports or browse both trace logs in split screen mode.

## Look for Application Problems

Now it is time to become acquainted with the trace displays and learn how to use them for application tuning.

There are two options on the Primary Option Menu that display trace data:

- Option 4, TRACES

OPTION ==> 4 TRACES - Current Application Traces

This option lists all currently active or complete trace requests. You have already viewed this option when starting your trace and checking that it started correctly. From this option you can see the trace data as it is being collected. And even after the trace is complete, it stays available here until you purge the trace.

Trace data is posted to the display buffer only when the accounting record is written. If you are tracing a longer running thread (detail), you can view the detail events as they occur with the UTRAC display. (Select the current USERS analyzer display, line select the detail display (DUSER) for your thread, and expand to UTRAC.)

When tracing long-running threads, logging is recommended so that data isn't lost because of buffer shortages. You should also browse the logs to view the data (see next option).

Press **PF3** to return to the Primary Option Menu.

- Option 5, HISTORY TRACES

OPTION ==> 5 HISTORY TRACES - Historical Trace Data Sets

Traces that are logged are also always accessible through this option that lists all known trace log data sets. You can view your trace when it is running, after it is complete, and even after you have purged the original trace request—as long as it is worthwhile to keep the data set. This makes it easy to compare different tests, even a month or two apart, without keeping stacks of paper.

From the log you can recall the data online or print selected reports. The trace log data sets are displayed in descending order by date and time, as shown in [Figure 68](#), so you can easily find a newly created log. If you are looking for an older log, you can sort the display by any of the columns.

*Logged Traces*

```

BOOLE AND BABBAGE ----- HISTORY TRACES ----- PERFORMANCE MGMT
COMMAND ==>                                     TGT ==> DB2G
                                           TIME -- 09:36:34  SCROLL ==> CSR
COMMANDS: SORT, LOCATE, NEW, STOP, START, TYPE
LC CMDS:  S (SELECT), W (SHOW), P (PRINT), D (DELETE), E (RESET)
           V (VERIFY), N (NEW), A (ARCHIVE), F (FREE)

DIRECTORY:   CIR4.LL1X.TRACEDI R
ENTRIES USED: 1,209  FREE:   201

LC  DATE-----TIME  TRACEID  TITLE                                USERID  TGT  STAT  ACTV
00/05/29 22:00  LEOTSTO1  LEOS DETAIL TRACE                    CIR4    DB2G  INV
00/05/11 21:00  DET1      WORKLOAD DETAIL 1                    CIR2    DB2G  USED  READ
00/05/01 00:00  THRDHIST  THREAD HISTORY                        BABUSERS DB2G  UPDAT WRT
    
```

Figure 68. History Traces Application

Figure 69 summarizes many of the ways you can navigate among the trace displays accessed from either Current Traces, Option 4, or History Traces, Option 5. It is included here for your reference.

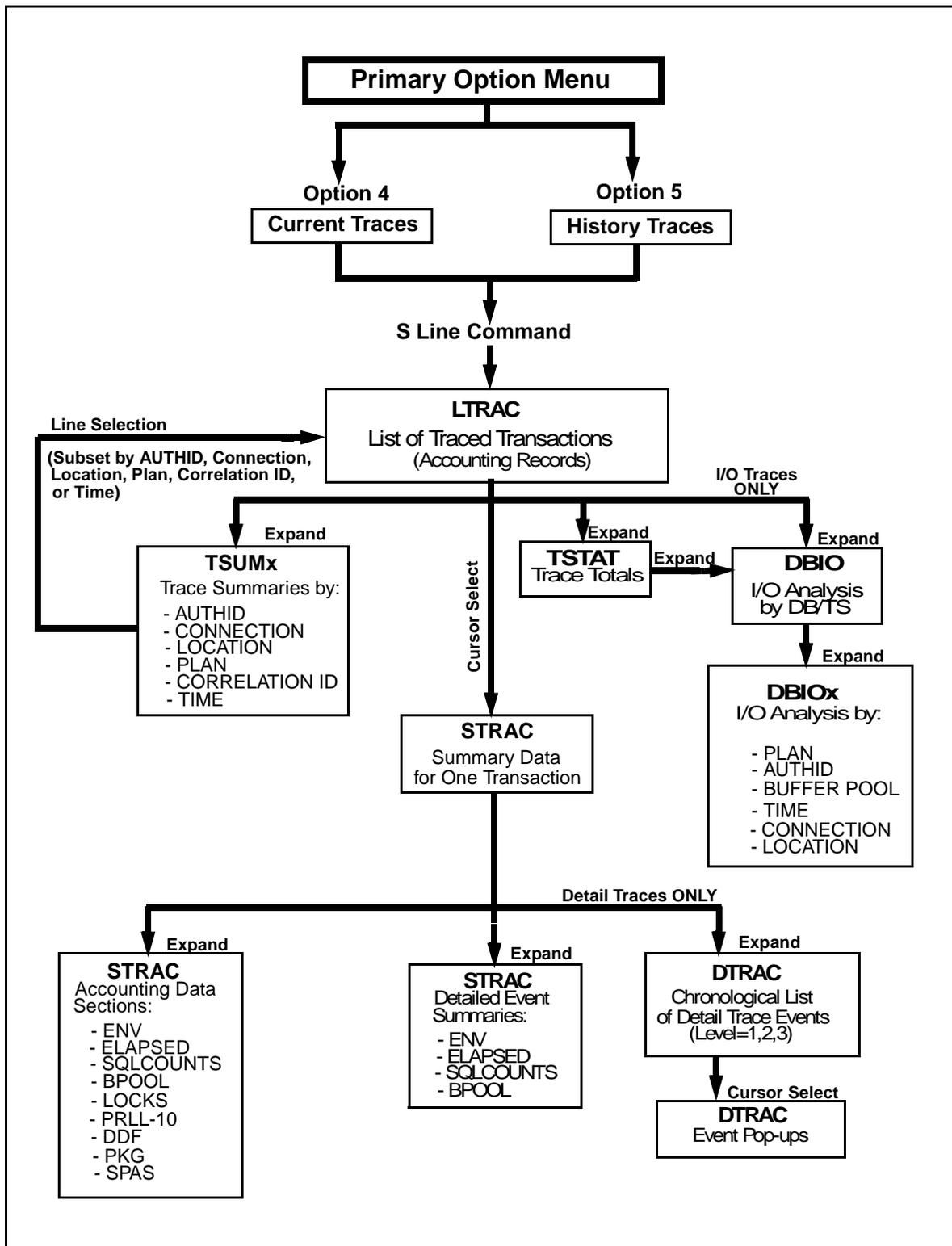


Figure 69. Trace Display Service Access

## All Trace Entries (DB2 Accounting Records)

Whether selected from Option 4 or 5, all the trace displays are the same, except for a few minor differences. So we will continue here with Option 5.

To view all trace entries in a trace log data set:

- Sort by USERID.

COMMAND ==>  **SORT US**

- Locate your user ID (like ISPF LOCATE).

COMMAND ==>  **L user id**

- Select the trace log data set that you just created.

LC (Line Command)  
**S** (for Select)

The first panel of trace data is displayed, as shown in [Figure 70](#).

*Expand from  
 Any Entry  
 for More  
 Information*

BOOLE AND BABBAGE----- DB2 TRACE ENTRIES ----- RX AVAILABLE									
SERV ==> LTRAC		INPUT 14: 11: 09		INTVL=> 3		LOG=> N		TGT==> DB2G	
PARM ==> PBCRO2						ROW 1 OF 24		SCROLL=> CSR	
EXPAND: MON(WKLD), TOTALS, I/O-DB/TS, HISTORY				ENTRIES IN DATASET 1 - 132					
AUTH, CONNECT, PLAN, TIME, LOC, LINESEL(STRAC)									
11MAY00 PBCRCP - SQL/SCAN/I/O									
END TIME	PLAN	AUTHID	CONNECT	ELAPSED	CPU	# STMTS	GETPAGE	REASON	
09: 17: 52. 76	DSNTIA21	CI R8X	BATCH	13 s	268 ms	4	459	OK TRM	
09: 18: 10. 33	DSNTIA21	CI R8X	BATCH	11 s	220 ms	12	168	OK TRM	
09: 18: 32. 78	DSNTIA21	CI R8X	BATCH	8, 768 ms	218 ms	12	162	OK TRM	
09: 18: 55. 13	DSNTIA21	CI R8X	BATCH	9, 225 ms	214 ms	12	162	OK TRM	
09: 19: 19. 80	DSNTIA21	CI R8X	BATCH	11 s	216 ms	12	162	OK TRM	
09: 19: 44. 59	DSNTIA21	CI R8X	BATCH	11 s	216 ms	12	162	OK TRM	
09: 20: 09. 28	DSNTIA21	CI R8X	BATCH	11 s	217 ms	12	162	OK TRM	
09: 20: 34. 83	DSNTIA21	CI R8X	BATCH	11 s	215 ms	12	162	OK TRM	
09: 21: 00. 54	DSNTIA21	CI R8X	BATCH	12 s	218 ms	12	162	OK TRM	
09: 21: 24. 18	DSNTIA21	CI R8X	BATCH	10 s	215 ms	12	162	OK TRM	
09: 21: 47. 62	DSNTIA21	CI R8X	BATCH	10 s	217 ms	12	162	OK TRM	
09: 22: 05. 21	DSNTIB21	CI R8X	BATCH	1, 117 ms	112 ms	49	59	OK TRM	
09: 22: 16. 03	DSNUTIL	CI R8X	UTILITY	2, 907 ms	163 ms	0	64	OK TRM	
09: 22: 17. 93	DSNUTIL	CI R8X	UTILITY	1, 886 ms	107 ms	0	61	OK TRM	
09: 22: 36. 33	DSNTIA21	CI R8X	BATCH	10 s	204 ms	4	357	OK TRM	
09: 22: 43. 14	DSNTIB21	CI R8X	BATCH	1, 981 ms	73 ms	49	32	OK TRM	
09: 22: 59. 79	DSNTIB21	CI R8X	BATCH	1, 439 ms	69 ms	49	30	OK TRM	

Figure 70. DB2 Trace Entries Display (LTRAC)

LTRAC lists all threads (trace entries) in chronological sequence, with the newest at the bottom. Each entry corresponds to a completed transaction, query, or batch job. There is one entry for each DB2 accounting record. In addition to the thread identifiers (plan, authid, connection), some of the most important performance indicators are shown.

- Use the scroll keys **PF7** and **PF8** to scroll back and forth in the list.

ENTRIES IN DATASET on line 4 tells you how many threads were traced. (This line says ENTRIES IN BUFFER from a current trace.) Compare the values for ELAPSED, CPU, #STMTS, and GETPAGE and look for high activity.

- Tab to any entry and press **Enter** to see more detailed information.

## Data for One Thread

The Summary Trace Entry display, STRAC, shows summary data for this thread. There are complete activity statistics from the DB2 accounting record. For a detail trace, there also are summaries of the captured detail events, such as SQL statements.

The most critical information is summarized in the base section, shown in [Figure 71](#). It includes

- Identifiers
- Completion status
- Commits and rollbacks
- A runtime graphic analysis of elapsed and CPU times
- The most important activity counts
- Key indicators of failures or possible problems

*Most Critical Information*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT   15:14:16 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> , SEQ=2              ROW 1 OF 129 SCROLL=> CSR
EXPAND: MON(WKLD), DETAIL, CURRENT              HIST TGT--- DSN2
ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
SUMMARIES: SQL, SCANS, IO/LOCK, SORTS
STOP. .... 07MAY 13:35:20.20 PLAN. .... DSN8CCO TYPE. .... ALLIED
START. .... 07MAY 13:35:20.17 AUTHID. .... T127 CONNECT. .... LRTEST1/CICS
ELAPSED. .... 33 ms ORIG PRIM AUTH. .... T127 CORR ID. .... GTOOPROB
TERM. .... NORMAL/APPL END COMMITS. .... 1 ROLLBACKS. .... 0
-----
RUNTIME ANALYSIS  IN DB2      IN APPL.      TOTAL      %IN DB2(=)      TOTAL(*)
-----
ELAPSED TIME          14 ms          19 ms          33 ms |=====*****|
CPU TIME              12 ms          1,552 us        13 ms |=====*|
DB2 WAIT TIME         0 us
-----
ACTIVITY ----- KEY INDICATORS -----
TOTAL SQL. .... 12          SQL: SELECT= 1, FETCH= 0
GETPAGES. .... 21          SQL: DYNAMIC(PREPARE)= 3
SYNC READS (PRL=00) .... 0          I/O RSP: SYNC= 45 ms, ASYNC= 107 ms
PREFETCH PAGES READ. .... 1
UPDATES/COMMIT. .... 0.0
BFR HIT RATIOS: ... VP=100%, HP= 0%
    
```

Figure 71. Summary Trace Entry Display (STRAC)—Base Section

## View DB2 Accounting Data

You can scroll down or expand to sections showing complete details on SQL statements by type, lock counts, buffer activity, and parallel I/O and package accounting:

1. Select **ELAPSED** in the ACCOUNTING EXPAND line.

If you run DB2 with Accounting Trace 2 or 3 active, an Elapsed Time Analysis section is displayed, as shown in [Figure 72](#).

*Why Are You Waiting?*

```

- - - - - ELAPSED TIME ANALYSIS (ACCTG CLASSES 2,3 ONLY) - - - - -
CATEGORY          #EVENTS  AVG/EVENT  ELAPSED  %TOTAL
-----
ELAPSED TIME
IN DB2              00: 12: 39  99.86  *****
IN APPLICATION      1,038 ms   0.13   <
-- TOTALS--        00: 12: 40  100.00  *****
WAITS IN DB2 (LOCAL)
LOCK/LATCH          0         0 us    0 us    0.00
I/O WAIT            19        21 ms   392 ms  0.05  <
LOG WRITE I/O       0         0 us    0 us    0.00
OTHER READ I/O      0         0 us    0 us    0.00
OTHER WRITE I/O     0         0 us    0 us    0.00
UNIT SWITCH EVENTS
.. COMMIT/ROLLBK    0         0 us    0 us    0.00
.. OPEN/CLOSE        6      8,965 ms  54 s    7.07  *
.. SYSLGRNG          1         700 ms   700 ms  0.09  <
.. DATASPACE MGR     4         243 ms   973 ms  0.12  <
.. OTHER              0         0 us    0 us    0.00
ARCH. LOG(QIS)      0         0 us    0 us    0.00
ARCH. READ(TAPE)    0         0 us    0 us    0.00
DRAIN LOCK          0         0 us    0 us    0.00
CLAIM RELEASE       0         0 us    0 us    0.00
PAGELATCH CONT.     0         0 us    0 us    0.00
SPAS SERVER TCB     0         0 us    0 us    0.00
WAITS IN DB2 (GLOBAL)
LOCKS               0         0 us    0 us    0.00
MSG. PROCESSING     0         0 us    0 us    0.00
--- TOTAL WAITS---  30      1,862 ms   56 s    7.34  *
*NOT ACCOUNTED      00: 11: 44  92.54  *****

```

Figure 72. STRAC Elapsed Time Analysis (ELAPSED) Section

The graph tells you at a glance where the most time is being spent—and what you should concentrate on in tuning: more time in the application or in DB2; if in DB2, are the times for I/O, prefetch reads, or lock waits unusually high?

2. Select **BPOOL** in the ACCOUNTING EXPAND line.

This section provides a complete summary of activity for each buffer pool accessed plus totals, as shown in [Figure 73](#).

*Buffer Pool Problems?*

----- BUFFER POOL ACTIVITY -----			
ACTIVITY	TOTAL	BP0	BP3
GETPAGES.....	30	25	5
SYNC READS.....	25	15	10
GETPAGES/READIO...	1.2	1.7	0
COND. GP FAILURES..	0	0	0
SEQ. PREFETCH REQS.	0	0	0
LIST PREFETCH REQS.	0	0	0
DYNAMIC PREFETCHES.	2	0	2
ASYNC PAGES READ...	0	0	0
PAGES/PREFETCH REQ.	0.0	0.0	0.0
PAGE UPDATES.....	3	2	1
IMMEDIATE WRITES...	0	0	0
HP SYNC READS.....	0	0	0
HP SYNC READ FAIL..	0	0	0
HP ASYNC PAGES READ	0	0	0
HP WRITES.....	0	0	0
HP WRITE FAILURES..	0	0	0
----- GLOBAL BUFFER POOL (DATA SHARING ONLY) -----			
CF READS (BUFFER CROSS INVALIDATION)			
- DATA RETURNED...	2	-	2
- R/W INTEREST.....	1	-	1
- NO R/W INTEREST..	1	-	1
CF READS (DATA NOT IN BUFFER POOL)			
- DATA RETURNED...	2	-	2
- R/W INTEREST.....	1	-	1
- NO R/W INTEREST..	1	-	1
CF- WRITE (CHGD PGS)	2	-	2
CF- WRITE (CLEAN PG)	1	-	1

Figure 73. STRAC Buffer Pool (BPOOL) Section

The GETPAGE / READ I/O ratio can give you a good indication of synchronous READ efficiency, which directly affects thread elapsed time. However, you should also check the prefetch requests, since some I/O may be occurring asynchronously.

3. Select each of the expand buttons shown in the ACCOUNTING line.

This is all the data you would see with a low-overhead summary trace.

- The package accounting section is available only if DB2 accounting class 7 is active, and shows a breakdown of elapsed, CPU, and wait times (class 8) per package/DBRM, as shown in Figure 74.

*Select a Package for More Information*

```

- - - - - PACKAGE / DBRM OVERVIEW (ACCTG CLASSES 7, 8 ONLY) - - - - -
PACKAGE/DBRM      NO.  SQL  CPU TIME  WAIT TIM  ELAPSED  %TOTAL
                   NO.  SQL  CPU TIME  WAIT TIM  ELAPSED  ELAP.
-----
RXSEL1M           5    23 ms   611 ms   860 ms   9.78 | *
RXSEL2M           6    21 ms   514 ms   538 ms   6.12 | *
RXSEL3M          24    38 ms   193 ms   250 ms   2.84 | <
RXSEL4M           8    22 ms   765 ms   828 ms   9.43 | *
RXSEL5M           9    24 ms   298 ms   337 ms   3.83 | <
RXSEL6M          10    25 ms   363 ms   408 ms   4.64 | <
RXSEL7M          11    26 ms   179 ms   211 ms   2.40 | <
RXSEL8M          12    28 ms   270 ms   299 ms   3.40 | <
RXSEL9M          13    27 ms   288 ms   351 ms   3.99 | <
RXSELAM          14    28 ms   256 ms   286 ms   3.25 | <
    
```

Figure 74. STRAC Package/DBRM Overview (PKG) Section

Select one package to see further details in a pop-up display.

- Press **PF3** to return to STRAC.

## View Detail Event Summaries (Detail Trace Only)

Since you started a detail trace, there are several more sections with data summarized from detail event records. You don't have to collect and analyze each type of data separately, the MAINVIEW for DB2 trace does it all for you. You can keep scrolling to see all the data, but there is a quicker way to select just the data you want.

### SQL Statement Summary

To view summary data for each SQL statement:

1. Tab through the SUMMARIES EXPAND line to the **SQL** button and press **Enter**.

The SQL Summary is displayed, as shown in [Figure 75](#), with important statistics for each SQL statement, showing totals and averages across all executions. If the plan contains multiple DBRMs, the statements are sorted by package/DBRM (program).

*Check Each  
SQL Statement*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT  14: 27: 16  INTVL=> 3  LOG=> N  TGT==> DB2G
PARM ==> PBCRO2, SEQ=5, SQL, SORT=PGM                ROW 1 OF 18  SCROLL=> CSR
EXPAND: MON(WKLD), DETAIL, HISTORY
ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
SUMMARIES: SQL, SCANS, IO/LOCK, SORTS

- - - - - SQL SUMMARY (DETAIL TRACE ONLY) - - - - -

  STMT      AVG.      %      AVG.      %      SORT  -- PAGES SCANNED --
  TYPE  STMT  COUNT  ELAPSED  ELAP  CPU    CPU  RECS  INDX DATA WORK  REF
-----
SELECT  3228      1    25 ms   2.7   1,930 us  0.3    0     2    1    0    0
SELECT  3347      2    11 ms   2.4   4,047 us  1.5    0    11    4    0    0
OPEN    3565      6    172 us  0.1    170 us  0.2    0     0    0    0    0
FETCH   3578     11   1,986 us  2.3    782 us  1.6    0    19    3    0    0
CLOSE   3664      6    192 us  0.1    149 us  0.2    0     0    0    0    0
SELECT  3671      6   1,776 us  1.1   1,091 us  1.2    0    12    0    0    0
SELECT  3283      1   2,643 us  0.3   1,888 us  0.3    0     1    0    0    0
OPEN    3299      1    109 us  0.0    107 us  0.0    0     0    0    0    0
FETCH   3313      2    907 us  0.2    855 us  0.3    0     1    0    0    0
CLOSE   3334      1    142 us  0.0    141 us  0.0    0     0    0    0    0
OPEN    3456      1    693 ms  74.1   414 ms  74.9    8    23 1187    2    0
FETCH   3468      5    440 us  0.2    322 us  0.3    0     0    0    6    0
SELECT  4803      1   1,269 us  0.1   1,121 us  0.2    0     2    0    0    0
PGM: P025D100      44                83.9                81.0    8    71 1195    8    0

SELECT  1239      1   1,040 us  0.1   1,040 us  0.2    0     2    1    0    0
OPEN    1263      1     66 ms   7.1     47 ms   8.6   172    13    6    9    0
FETCH   1273     87    201 us  1.9    177 us  2.8    0     0    0    2    0
CLOSE   1324      1    250 us  0.0    250 us  0.0    0     0    0    0    0
OPEN    1333      1     15 ms   1.6     14 ms   2.6    7     6    4    5    0
FETCH   1343      5    254 us  0.1    254 us  0.2    0     0    0    2    0
SELECT  1375      3   1,746 us  0.6   1,165 us  0.6    0     0   12    0    0
CLOSE   1393      1    126 us  0.0    126 us  0.0    0     0    0    0    0
PGM: P025D200    100                11.4                15.0   179   21  23   18    0

SELECT   389      1     38 ms   4.2     18 ms   3.4    0    10    3    0    0
OPEN     482      1    135 us  0.0    134 us  0.0    0     0    0    0    0
FETCH    489     10    459 us  0.5    302 us  0.5    0     1    0    0    0
PGM: P025D300    12                4.7                3.9    0    11    3    0    0

** TOTALS ***      156                187   103 1221   26    0
    
```

Figure 75. STRAC SQL Summary Section

2. Scan the PERCENT ELAPSED column.

The percent tells you how much this statement is contributing to the total thread elapsed time. An average may be high, but if only executed a few times, the statement may not be worthwhile tuning.

3. Scan the SORT RECS column to see which statements invoked a sort.

Although EXPLAIN tells you a sort will be used, it can't tell you whether many rows will be selected and sorted, or just a few. This column does. (Of course, as always, you must adjust this by any differences between your test and production tables.)

4. Scan the PAGES SCANNED - INDX column.

This tells you whether an index was accessed and how many pages were scanned. If this value doesn't meet your expectations, there is more information on index accesses a little further along.

5. Check for referential integrity processing. See the PAGES SCANNED - REF column.

6. Sort the display by PAGES SCANNED - DATA.

```
PARM ==> traceid, SEQ=nn, SQL, SORT=PD
```

7. Now sort the display by PAGES SCANNED - DATA within Program.

```
PARM ==> traceid, SEQ=nn, SQL, SORT=PDP
```

The first characters of the column header are used to request a sort. Adding P as the third character keeps all the statements for each DBRM together. All options are defined in the HELP panels (PF1). (The sort is supported only when you expand to the section, not when you scroll to it.)

- Place the cursor on one of the SQL statement lines and press Enter.

This pop-up display, shown in [Figure 76](#), shows the complete statistics for that statement, displayed as averages per execution. Where the single line shows total counts of pages scanned, the pop-up display shows averages not only for pages but also rows accessed.

*Evaluate SQL  
Predicates*

BOOLE AND BABBAGE-----		DETAIL TRACE ENTRY -----		RX AVAILABLE	
SERV ==>	STRAC	INPUT	10: 22: 51	INTVL=>	3 LOG=> N TGT==> DB2G
PARM ==>	TEST3, SEQ=000017, SQL	ROW	1 OF	18	SCROLL=> CSR
EXPAND:	<b>SQLTEXT(EXPLAIN)</b>				
STATEMENT:	350 SELECT	NUMBER OF EXECUTIONS:		2	
PLAN:	DSNESPRR	ELAPSED:	AVERAGE 2, 794 us	TOTAL 5, 588 us	
PROGRAM:	DSNTIAUL	CPU:	AVERAGE 1, 271 us	TOTAL 2, 543 us	
LOCATI ON:	DB1D				
PACKAGE:	SAJUYH2I	----- AVERAGES -----			
		INDEX	SEQ- DATA	SEQ- WORK	
		-----	-----	-----	
	ROWS PROCESSED ALL TYPES	4	2	0	
	ROWS PROCESSED CORRECT TYPE	4	2	0	
	ROWS QUALIFIED BY DM (STAGE 1)	3	0	0	
	ROWS QUALIFIED BY RDS (STAGE 2)	0	0	0	
	ROWS INSERTED	0	0	0	
	ROWS UPDATED	0	0	0	
	ROWS DELETED	0	0	0	
	PAGES SCANNED	8	2	0	
	REFERENTIAL INTEGRITY PROCESSING:				
	PAGES SCANNED	0	0	0	
	ROWS DELETED/SET NULL	0	0	0	

Figure 76. STRAC SQL Statement Pop-Up Display

These are the statistics that allow you to evaluate the SQL predicates. For example, ROWS QUALIFIED BY RDS are caused by Stage 2 predicates and are more expensive than rows qualified in the first stage by the Data Manager (ROWS QUALIFIED BY DM). Of course, there may be variations per execution depending on host variable values. You will see how to find the statistics per single statement execution later.

- If RxD2 is installed and this is a static SQL statement, the **SQLTEXT(EXPLAIN)** expand button is highlighted. You can select this to view the SQL text from the catalog and access available EXPLAIN data from a PLAN\_TABLE or do a dynamic EXPLAIN. This is shown later in [Chapter 6, “Using RxD2 with MAINVIEW for DB2”](#) on page 111.
- Press **PF3** to return to STRAC.

**Table Space / Index Space Scans**

To view table space and index space scans:

1. Tab through the EXPAND line to the **SCANS** button and press **Enter**.

Here you can see a summary of all scans by table space and index space, as shown in [Figure 77](#).

*Review  
Table Spaces  
and  
Index Access*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT 14:27:46 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> PBCRO2, SEQ=5, SCAN, SORT=DA          ROW 1 OF 11 SCROLL=> CSR
EXPAND: MON(WKLD), DETAIL, HISTORY
ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
SUMMARIES: SQL, SCANS, IO/LOCK, SORTS

----- DATABASE SUMMARY (DETAIL TRACE ONLY) -----

```

DATABASE	PAGESET	OBID	----- INDX -----			----- DATA -----			----- WORK -----		
			#SCAN	#PAGE	#ROWS	#SCAN	#PAGE	#ROWS	#SCAN	#PAGE	#ROWS
DD028	S021FT10	8	16	32	193	12	9	185	0	0	0
DD028	S021FT20	13	94	18	110	2	4	16	0	0	0
DD028	S021FT30	18	0	0	0	3	12	39	0	0	0
DD028	S021FT40	23	3	3	360	0	0	0	0	0	0
DD028	S021PR10	28	6	12	7	6	1181	44566	0	0	0
DD028	S021PR30	33	33	36	815	11	14	298	0	0	0
DD028	S021UI10	40	1	2	1	1	1	1	0	0	0
DSNDB07	DSN4K01	3	0	0	0	0	0	0	3	6	0
DSNDB07	DSN4K02	5	0	0	0	0	0	0	1	2	0
DSNDB07	DSN4K03	7	0	0	0	0	0	0	2	4	0
****	TOTALS	*****	153	103	1486	35	1221	45105	6	12	0

Figure 77. STRAC Database Summary Section

This data is available only if you requested SCANS on the trace request. It allows you to review the number of rows and pages accessed per page set, and also see which indexes were used.

**Lock and I/O Events**

To analyze lock and I/O problems:

1. Tab through the EXPAND line to the **IO/LOCK** button and press **Enter**.

This display, shown in [Figure 78](#), is a summary of all lock and I/O activity by this application, shown by table space and index space. The lock data is available with any detail trace. The I/O data is available only if you requested I/O events on the trace request, which can be expensive, depending on the application characteristics.

*Analyze Lock  
and  
I/O Problems*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT  14:27:53 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> PBCRO2, SEQ=5, IOLOCK, SORT=DP          ROW 1 OF 11 SCROLL=> CSR
EXPAND: MON(WKLD), DETAIL, HISTORY
        ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
        SUMMARIES: SQL, SCANS, IO/LOCK, SORTS

- - - - - DATABASE LOCK AND I/O SUMMARY (DETAIL TRACE ONLY) - - - - -

----- LOCKS ----- ----- SYNC. I/O ----- ASYNC I/O
DATABASE PAGESET  MAX.  SUSP.  TM/OUT  READ WRITE  ELAPSED  AVG. ELAP  READ PAGES
-----
B1CODB  BB040A12    0    0    0      1    0    21 ms   21 ms    0    0
B2CODB  BB051012    2    0    0      1    0    18 ms   18 ms    0    0
B3CODB  BB09TS02    3    0    0      0    0     0 us    0 us    0    0
**** TOTALS ****    5    0    0      2    0    39 ms   19 ms    0    0
    
```

Figure 78. STRAC Database Lock and I/O Summary Section

For traces by AUTHID, where SYSOPR is included, you will also see counts for prefetch read requests and the number of pages read for this thread. The other counts are all for synchronous I/O within the application. In other words, the ELAPSED time shown here is part of the total thread elapsed time. Reducing the total number of synchronous I/Os has a direct effect on performance.

We will show you later how to relate these I/Os to the packages/DBRMs and even the SQL statements that caused them. (See [“Application I/O Analysis”](#) on page 86.)

### Avoiding Expensive Lock Event Traces

A lock trace is usually only requested to analyze a specific locking problem, since it causes a great deal of DB2 overhead and also generates many trace records for MAINVIEW for DB2 to store online and/or log. Therefore, before you decide to trace LOCK events, you should analyze the following detail trace events available in DTRAC (shown next):

**LOCK-SUMMARY (display LEVEL=2)**

This event appears near the end of the thread processing. It shows the maximum number of page locks held and the highest lock state for each page set (table/index space) accessed.

**LOCK-TMO (display LEVEL=1)**

This event is shown for any thread terminated by either a timeout or deadlock. It shows the page set involved and identifies the first holder of the lock that was in contention, as well as the state of the lock (for example, held exclusive). It is followed by either a DEADLOCK or LOCK-TMO-DET event (display LEVEL=2) that gives complete details.

**LOCK-SUSP (display LEVEL=3)**

This event is shown for all lock suspensions. It shows the elapsed time suspended, the page set, the type of entity locked (for example, a data page, index tree, and so on), page number if applicable, and lock state.

### Sort Information

To evaluate the impact of sorting:

1. Tab through the EXPAND line to the **SORTS** button and press **Enter**.

A Sort Summary of all the sort activity for this plan execution is displayed, as shown in [Figure 79](#).

*Evaluate  
Impact  
of Sorting*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT  14:27:16 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> PBCRO2, SEQ=5, SORTS                                ROW 1 OF 18 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
          ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
          SUMMARIES: SQL, SCANS, IO/LOCK, SORTS

----- SORT SUMMARY (DETAIL TRACE ONLY) -----
                                     SORT      SORT      RECORD
                                     ELAPSED   RECORDS   SIZE
-----
NUMBER OF SORTS..... 6  AVERAGE   135 ms      31      353M
AVG # WORK FILES.... 1.3  MAXIMUM   263 ms      86     2,118M
                               MINIMUM   263 ms       3         0
                               TOTAL     263 ms     187      N/A
***** END OF DATA *****
    
```

Figure 79. STRAC Sort Summary Section

When scrolling through STRAC, this section is at the end of the display. The SORT ELAPSED time allows you to evaluate the impact of sorting on the total thread elapsed time. Since there may be multiple sorts, the values are shown as averages, maximums, minimums, and totals.

## Look for Application Problems

Now you have seen all the summarized data for one thread by browsing through an STRAC (Summary Trace Entry) display. Even with a low-overhead summary trace, the complete accounting data, elapsed time analysis, buffer pool usage analysis, and DDF information is available. The other summarized data depends on the type of events you decide to trace: SQL, SCANS, IO, LOCKS.

2. Press **PF3** several times until you return to the initial STRAC display.

## Another Thread

If you have traced several thread executions, you may now want to view another thread. Look at the PARM field. The SEQ=nnnnnn field shows you the sequence number of this trace entry (accounting record plus summarized details) and allows you to navigate between entries.

To view another thread:

1. Press **PF10** to scroll to the next chronological entry.
2. Press **PF11** to scroll back to the previous entry.
3. Select the **SQLCOUNTS** expand button to view the summary of all SQL statements; then press **PF10** several times.

The display shows the SQL counts for the following threads. This is an easy way to compare particular statistics (such as SQL, elapsed times, or GETPAGEs) for several threads.

## Detail Events

There is another level of trace data available for a detail trace. It shows all the traced events for a thread in chronological sequence. To see the detail event statistics:

1. Tab through the EXPAND line to the **DETAIL** button and press **Enter**.

The DTRAC (Detail Trace Entry) display allows you to follow the sequence of events during thread processing, as shown in [Figure 80](#).

*Follow the Sequence of Events*

BOOLE AND BABBAGE----- DETAIL TRACE ENTRY ----- PERFORMANCE MGMT				
SERV ==>	DTRAC	INPUT	10:21:48	INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==>	TEST3, SEQ=000017, LEVEL=2	ROW	1 OF	35 SCROLL=> CSR
EXPAND:	LINESEL(DETAIL), HISTORY			
START:	09:17:11 AUTH: JEK1	PLAN:	DSNTIB21 CORR: DB221REQ	CONN: BATCH
EVENT	AT	ELAPSED	CPU	DETAIL
CREATE-THD	0.000	45 ms	4,492 us	
PLAN-ALLOC	0.045			ISOLATION=CS ACQ=USE REL=COMMIT
PKG-ALLOC	0.065			*DSNESPSC ISO=CS ACQ=USE REL=COMIT
PREPARE 350	0.069	1,361 ms	26 ms	*RC( 0) C=DT D/X PS( 10)
BIND-TEXT	0.070			*TYPE=DYNAMIC TEXT=SELECT * FROM DS+
EXPLAIN	0.078			*PLAN=DSN8I C22 COST(4.6)
EDM-REQ	1.302	123 ms	3,964 us	DB=00000258
OPEN 524	1.489	204 us	202 us	*RC( 0) C=DT
FETCH 532	1.489	2,058 ms	6,247 us	*RC( 0) C=DT D/X PS( 2)
OPEN-TS	2.235			DB=DSN8D21A TS=DSN8S21D
OPEN-TS	3.352			DB=DSN8D21A TS=XDEPT3
FETCH 532	3.549	35 ms	441 us	*RC( 0) C=DT D/X
FETCH 532	3.615	354 us	353 us	*RC( 0) C=DT D/X
FETCH 532	3.616	335 us	335 us	*RC( 0) C=DT D/X
FETCH 532	3.620	354 us	353 us	*RC( 0) C=DT D/X
FETCH 532	3.621	386 us	387 us	*RC( 0) C=DT D/X
FETCH 532	3.672	386 us	387 us	*RC( 0) C=DT D/X
FETCH 532	3.673	337 us	337 us	*RC( 0) C=DT D/X

Figure 80. Detail Trace Display for a Transaction (Level 2)—DB2 Requests

The number and type of events shown depend on how much you decided to trace (more events, more overhead). The simplest detail trace captures just the major events and exceptions in the life of the thread. This trace also has SQL, SCANS, and IO.

2. Look at the first event, CREATE-THD.

Events like this, with start and end trace records, are matched and displayed on one line with relative start time, elapsed, and CPU times. Events that occur within another event (like I/O for a scan) are indented.

3. Look at the PARM field.

The SEQ number is the same as that shown on the Summary display (STRAC). But now there is a new parameter of **LEVEL=2**. This parameter allows you to widen or narrow your view of all the traced events. Level 2 shows you the major events (for example, the first event you see is probably PLAN-ALLOCATION) and SQL.

4. Change the level to 1 to see just the SQL statements.

PARM ==> t traceid, SEQ=nnnnn, LEVEL=1

Browse through the SQL statements to find one you want to know more about.

5. Tab to that statement and press **PF7** to scroll it to the top (SCROLL => CSR).

6. Change the level to 3 to see SCANS and I/O.

PARM ==> traceid, SEQ=nnnnnn, LEVEL=3

The events always remain in chronological sequence, and the event at the top of the screen remains there. Now you can see all of the events that occurred in processing the SQL statement—index scans, data scans, table space opens, sorts, dynamic SQL text and EXPLAIN data, I/O, and so on.

7. Move the cursor to an SQL statement and press **Enter**.

Here are the statistics for the single SQL statement, shown in [Figure 81](#), available in a pop-up display whenever you need them, but out of the way when browsing the events.

*Event Pop-Up*

*Access More Information*

```

BOOLE AND BABBAGE-----  DETAIL TRACE ENTRY  -----  RX AVAILABLE
SERV ==> DTRAC          INPUT    10:48:24  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> ALLNEW51, SEQ=4, LEVEL=2          ROW 1 OF 17  SCROLL=> CSR
EXPAND: . SQLTEXT(EXPLAIN)
START: 13:59:23 AUTH: BOLSMR3  PLAN: DSNTIA51  CORR: DB251REQ  CONN: BATCH
=====
EVENT          AT          ELAPSED      CPU    DETAIL
-----
CREATE    1817      0.475      14 s     62 ms *RC( 0) TABLESPACE DEMOS51D
=====
CURSOR:      DEMOS51D
LOCATION:     DB2H
ISOLATION:  READ STABILITY
REOPTIMIZ:  REOPTIMIZATION FOR INPUT VARS
PACKAGE:    SAJUYH2I
PROGRAM:    DSNTIAD
                INDEX  SEQ-DATA
                -----
ROWS PROCESSED ALL TYPES                6          6
ROWS PROCESSED CORRECT TYPE             12         6
ROWS QUALIFIED BY DM (STAGE 1)           6          0
ROWS QUALIFIED BY RDS (STAGE 2)          0          0
ROWS INSERTED                             0          1
ROWS UPDATED                              0          0
ROWS DELETED                              0          1
PAGES SCANNED                             17         14
REFERENTIAL INTEGRITY PROCESSING:
PAGES SCANNED                             0          0
ROWS DELETED/SET NULL                     0          0
    
```

Figure 81. DTRAC SQL Statement Pop-Up Display

Any event that has an asterisk at the beginning of the DETAIL column has a pop-up display. The detail column shows the most important data, but many trace records have much more information available. This data varies by event type. For example, for an SQL statement, you see the return code, cursor, indicators about data access (Data/Index/Work = D/X/W), and pages scanned.

You have already seen the SQL pop-up display; the same row/page statistics are available per scan. Some other important pop-up displays are

- BIND-TEXT            Text of dynamic SQL (also static SQL if BIND traced)
- EXPLAIN            EXPLAIN data for dynamic SQL (also from BIND)
- LOCK-SUMMARY       Summary of locks held per page set
- LOCK-TMO           Identification of lock holder causing a timeout/deadlock
- MULTI-INDEX        Statistics on RID list processing, success, or failure

8. Press **PF3** three times to return from DTRAC to the LTRAC list of thread entries.

## Summarized Data for All Threads

Now look at data that has been summarized for all the threads that have been traced:

1. In LTRAC, tab to the **TOTALS** expand button and press **Enter** to access the TSTAT display, as shown in [Figure 82](#).

*Trace Totals*

```

BOOLE AND BABBAGE----- TRACE STATISTICS ----- RX AVAILABLE
SERV ==> TSTAT          INPUT  14: 13: 18 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> THRDHIST, PLAN=DSNTIA22, TIME=1810- 1820      ROW 1 OF 37 SCROLL=> CSR
EXPAND:  MON(WKLD),  HISTORY
          ACCOUNTING: ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, SPAS, DDF
          SUMMARIES:  SCANS, I/O/LOCK, SORTS, I/O-DB/TS
----- SUMMARY STATISTICS - ALL TRACE ENTRIES-----
----- TERMINATIONS ----- ACTIVITY-----
FIRST END. . 22MAY 11. 07. 26. 38          TOTAL  AVERAGE  MAXIMUM  MINIMUM
LAST END. . 22MAY 12. 04. 39. 68          -----
NUMBER TRANS. . . . . 191  ELAPSED  02: 05: 18    39 s    00: 26: 40    539 us
COMMIT/ROLLBK. . . . . 188/. . . . . 3  ELP-DB2  00: 05: 17    1,662 ms  00: 01: 13    0 us
NORMAL TERM. . . . . 189  CPU      35 s     183 ms    2,332 ms    0 us
-- NEW USER. . . . . 2    CPU-DB2  23 s     121 ms    2,079 ms    0 us
-- DEALLOC. . . . . 137  WAITS   00: 04: 14    1,330 ms  00: 01: 13    0 us
-- APPL END. . . . . 0    SQL     3,926     20       513         0
-- RESIGNON. . . . . 0    GETPAGES 17,280     90       1,857       0
-- DBAT INACT. . . . . 50  SYNC RDS  424       2         70         0
-- I FI READ. . . . . 0    PFCH PGS  846       4         252        0
ABNORMAL TERM. . . . . 1    UPD/COMT  247       1         24         0
IN DOUBT TERM. . . . . 0    BFR HIT RATIOS: . . . . . VP= 93%, HP= 0%
----- KEY INDICATORS -----
TIMEOUTS = 5
BUFFER INCOMPLETE = 1
TOTAL DDL = 5
GRANTS / REVOKES = 7
SQL: SELECT= 209, FETCH= 2,248
SQL: INS= 155, UPD= 219, DEL= 87
SQL: DYNAMIC(PREPARE)= 247
I/O RSP: SYNC= 19 ms, ASYNC= 106 ms
LOCK SUSPENSIONS = 24
RID LIST PROCESSING USED = 314
    
```

Figure 82. Summary Statistics for All Trace Buffer Entries (TSTAT)—Base Section

The TSTAT (Trace Statistics) display provides a summary of all traced threads. Much of the data is similar to that shown for one thread in the STRAC display, so just browse through it using the **PF8** key or select the desired accounting data sections from the EXPAND line.

TSTAT can be useful in application tuning when you trace just one plan. It gives you an overview of the performance of the plan to see if you even need to go into further detail analysis of single thread executions. If the average values meet expectations and the maximum and minimum don't vary greatly, you may be able to stop here. Also, TSTAT summarizes the key indicator values to show whether any exceptional conditions, such as RID pool failures, have occurred. After all, the art of performance analysis and tuning is to spend your time where you get the biggest payback.

2. Press **PF3** to return to LTRAC.

## Application I/O Analysis

Earlier you looked at the I/O for one thread. Now look at the I/O performed for all the threads traced.

The best way to analyze the I/O workload is to use the series of DBIOx displays, available with an expand button from either LTRAC or TSTAT on an I/O trace, as follows:

1. Tab to the **I/O-DB/TS** expand button and press **Enter** to access the DBIO display, as shown in [Figure 83](#).

*Summarized  
I/O Events  
by Page Set*

```

BOOLE AND BABBAGE----- I/O Analysis- HT ----- PERFORMANCE MGMT
SERV ==> DBIO INPUT 11:02:14 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> , SORT=DB, TOTAL LINE 1 OF 19 SCROLL=> CSR
EXPAND: PLAN, AUTH, BPOOL, TIME, CONNECT, LOCATION
EXPAND: LINESEL(DBTS), CATALOG
OPTION: TOTAL, SYNC, ASYNC

11MAY00 CI R2. DB2G. JQSI0. MAY11. T1818. V01
DATA TABLE I/O I/O MAX AVG
BASE SPACE COUNT % IOWAIT IOWAIT
-----
--ms-- --ms-- 0 ... 20... 40... 60... 80
DSNDB01 DBD01 4 5.6 89 34 | ***** |
DSNDB01 DSNCT02 1 1.4 28 28 | ***** |
DSNDB01 DSNST01 1 1.4 17 17 | **** |
DSNDB01 SCT02 2 2.8 38 31 | ***** |
DSNDB01 SPT01 4 5.6 351 95 | ***** |
    
```

Figure 83. I/O Analysis by Database / Table Space Display (DBIO)

The first display (DBIO) summarizes the I/O events by database and table space. You can see the number and percent of I/O per table space and index space, as well as average and maximum I/O wait times. From here, you can use the expand buttons, line selection, and sort parameters to focus on the area of most interest, whether a particular table space, an application, or a specific time period.

2. Tab to **SYNC** on the OPTION line and press **Enter**.

Now only synchronous I/O are displayed. SYNC is now in the PARM field. TOTAL (the default) includes both synchronous and asynchronous I/O. ASYNC shows the prefetch I/O.

3. Tab to **PLAN** and press **Enter**.

Now the I/Os are summarized by plan.

4. Tab to the **I/O %** column and press **Enter** to sort the plans with the highest amount of I/O at the top of the list.

5. Line select one of the plans and press **Enter**.

The original display by page set is shown again, but is now qualified to show only the I/Os for the selected plan.

Press **PF3** to return to the DBIOP display.

6. Tab to **PKG/PGM** and press **Enter**.

The programs (DBRMs or packages) per plan causing the I/O are now identified.

7. Tab to **SQL** and press **Enter**.

This lowest level now identifies each SQL statement causing I/O, as shown in [Figure 84](#).

*SQL  
Statements  
Causing I/O*

```

BOOLE AND BABBAGE----- I/O Analysis- SQL Stmt ----- PERFORMANCE MGMT
SERV ==> DBIOS          INPUT    12:01:32 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> , SO=PL, SYNC          LINE    1 OF    3 SCROLL=> CSR
EXPAND: LI NESEL(DBIO), CATALOG, HI STORY
OPTION: TOTAL, SYNC, ASYNC

11MAY00                      DB2 APPLICATION TRACE
      PACKAGE/ SQL          I/O  I/O  MAX  AVG
PLAN  PROGRAM  STMT      COUNT  %  IOWAIT IOWAIT
-----
RXDB2  RXSEL2M  228          30 93.8  21464  1505 | ***** |
RXDB2  RXSEL4M  115           1  3.1    49    49 | ***** |
RXDB2  RXSEL9M  228           1  3.1    49    49 | ***** |
    
```

Figure 84. I/O Analysis by SQL Statement (DBIOS)

8. At any time you can return to LTRAC (press **PF3** four times from DBIOS). Select a single thread (for example, one with a high elapsed time) and look at the I/O per page set for that one execution summarized in the STRAC IO/LOCK section.
9. Select the STRAC **DETAIL** button to view the detail events. Change the display level to **3** to see I/O events.

You can now see each I/O in the chronological sequence in which it was executed, following the SQL statement that caused it, with elapsed and CPU times.

---

## Print a Trace Report

You have now run a trace and viewed it online. However, depending on the results, you may want to have a hardcopy for further analysis. Batch reports can also be valuable tools during an application review meeting.

**Note:** This section describes printing a trace, but you may also want to look at the accounting reports produced from SMF data, either from DB2 tables or directly from the extracted DB2 accounting records. See the *MAINVIEW for DB2 Performance Reporter User Guide* for more information.

To begin this exercise:

1. Press **PF3** several times until you return to the History Traces panel (Option 5).

This panel not only provides access to view the trace data online but also offers several line commands to simplify management of the trace log data sets:

- W** Show the options used for this trace
- P** Generate the JCL to print a batch report
- D** Delete this data set from the trace directory
- E** Reset the log data set for reuse
- V** Verify that the directory entry matches the data set contents
- N** Add a new data set to the directory (moved from another system)
- A** Archive the data set (only if an archive started task was specified)
- F** Free a data set currently being read

Usually you will need only W(SHOW), D(DELETE), and P(PRINT). We are going to concentrate now on P. You can try the others when you need those functions.

**Note:** The P option only works when running under ISPF.

## All Data per Traced Thread

To print a trace report showing all data per traced thread:

1. Select your trace log data set for print.

LC (Line Command)  
**P** (for PRINT)

The Batch Trace Print panel is displayed, as shown in [Figure 85](#), where you can specify options to print a batch report.

### Printing a Trace

```

BOOLE AND BABBAGE ----- BATCH TRACE PRINT ----- PERFORMANCE MGMT
COMMAND ==>>                                         TIME --- 15:00

Update job ==> N (Y/N - update job statement)          (END to edit JCL)
Title line 1 ==>
Title line 2 ==>

Data Selection:
From date ==> 10MAY2000  Time ==> 1412
To date ==> 11MAY2000  Time ==> 1125

PLAN ==>
AUTHID ==>
CONNECT ==>
CORR ==>
LOC ==>
DB2PKG ==>

REPORT SELECTION:
LTRAC ==> NO (YES/NO)
STRAC ==> NO (NO, ALL, SUMMARY, section1, section2. .)
DTRAC ==> NO (YES/NO)
POPUP ==> NO (NO/ALL/SQL/event1, event2, . . .)
TSUM ==> (A/C/L/P/T) SORT ==> (for any TSUMx)
TSTAT ==> NO (SUMMARY/ALL/NO) INTERVAL ==> 1H (for TSUMT only)
DBIO ==> (X/A/C/F/L/P/T/xx) IOSORT ==> (for any DBIOx)
IOINTVL ==> 10M (for DBIOT only)

```

Figure 85. Batch Trace Print Panel

2. If this is your first time using this option, you must update your job statements.

Update Job ==> **Y**

3. Press **Enter** to display a job statement data entry panel.

Fill in the required information and return using **PF3**.

4. Set the option to **N**.

5. Fill in the options for a report.

TITLE1 ==> **any title** Optional. Centered in first report header.  
TITLE2 ==> **any title** Optional. Centered in second report header.

Data Selection is optional. The selection fields can be used to reduce the amount printed from a long trace. The date and time fields are primed with the start and end date-time of the trace.

**REPORT SELECTION:**

LTRAC ==> N  
STRAC ==> **ALL** Print data per thread traced.  
DTRAC ==> N  
POPUP ==> N  
TSUM ==>  
TSTAT ==> N  
DBI O ==>

There are many different reports that can be generated, either singly or in combination. Browse "Printing a Trace" in Volume 2 of the *MAINVIEW for DB2 User Guide* for a full explanation of the options and to see sample reports. There is also a sample JCL member DZTBPRNT you can use when you want to generate multiple reports or print reports from SMF data. The control statements for several sample reports are provided in BBSAMP member DZJPnnnn, with comments to point out some of the most useful variations.

The reports are based on printouts of the online displays, so the options are selected using the names of these displays, like LTRAC, STRAC, DTRAC. In this tutorial we have chosen **STRAC=ALL** as the most useful report for application tuning. It shows the following for each thread traced:

- Basic **DB2 accounting record** data
- **Environmental Indicators** section
- **Elapsed Time Analysis** section
- **SQL Statement Execution Counts** section
- **Buffer Pool Usage Analysis** section, including Global Buffer Pools
- **Lock Activity** section, including Global Locks
- **I/O Parallelism** section
- **Routines** section (stored procedures and user-defined functions)
- **DDF Summary** section (if distributed work was done)
- **Package/DBRM Overview** section (if accounting trace 7 is active)

These sections are included for detail traces:

- **SQL Summary** section with summary statistics per SQL statement
- **Database Summary** section with scans per page set
- **Database Lock and I/O** section with locks and/or I/O data per page set
- **Sort Summary** section (if any sorts were performed)

6. Press **Enter** to validate your specifications.

These options are saved in your profile and used to initialize the fields the next time this panel is requested.

7. Press **PF3** to review the generated JCL in edit mode.

The options you specified are inserted into a pattern job DZJPTRAC in BBPROF. You may want to copy this JCL into your own UBBPROF profile data set and modify it.

8. Check the rest of the options.

There are many more print options than can be shown on the panel. The sample job contains a short description of these options. Scroll to the bottom to review them.

9. Submit the job.

**COMMAND** ==> **SUB**

Of course, if you prefer, you can **SAVE** the JCL for later execution, or even **CANCEL** it completely.

10. Press **PF3** to return to the Batch Trace Print panel. You can issue another request now if desired.

## Other Examples

When the job is completed, review the output. You may want to try other report combinations too. Here are a few examples:

- For an SQL statement summary per thread, sorted by Average CPU usage, followed by average SQL row processing statistics per statement:

**LTRAC** ==> **Y**  
**STRAC** ==> **SQL, SORTSQL=AC, SQLPOPUP**

- For a detail event trace per thread, with pop-up displays per SQL statement:

**LTRAC** ==> **Y**  
**DTRAC** ==> **Y**  
**POPUP** ==> **SQL**

- For SQL statement text and EXPLAIN data for all dynamic SQL executed or BINDs of static SQL:

**POPUP** ==> **(BIND-TEXT, EXPLAIN)**

- For a summary of I/O counts and wait times per database, table space, and plan (from an I/O trace):

**DBI O** ==> **XP**

- To summarize by plan, database, and table space:

**DBI O** ==> **PX**

There are also many formatting options you can adjust:

**NEWPAGE** Control page breaks

**WIDTH** Specify wide (133) or narrow (81) output

**LINECNT** Adjust the number of lines printed per page

**HEADING** Suppress headings

**MAXPAGES** Limit the amount of output to prevent an unexpectedly high print volume



---

## Chapter 5. Analyzing the DB2 Workload with Trace

These scenarios teach you how to use the trace displays and the sorting and selection features to pinpoint problem areas and exception threads.

In this practice session, you

1. Analyze recent workload history through Option 6—GRAPH.
2. Review past history from the trace logs.
3. Analyze DB2 I/O per table space.
4. Print workload reports for offline review.

**Note:** This practice session takes approximately one hour to complete.

## Analyze Recent Workload History

Sometimes you may need to go back in time to analyze recent history of your DB2 workload; for example:

- To identify system and application exceptions and relate the problems to the threads that caused them
- To analyze the performance of a specific application
- To develop an understanding of the workload profile and spot deviations

## Graphic Display of Thread History

MAINVIEW for DB2 sets up a summary trace of the complete DB2 workload that is started automatically and runs continuously. This trace captures the DB2 accounting records and is inexpensive to run. The system administrator may have chosen to log the data, in which case a much longer time span will be available for online recall.

To view a graphic display of this recent thread history:

1. From the Primary Option Menu, select the **GRAPH** option.

OPTION ==> **6**

This panel displays *thread history summarized by 10-minute intervals*, sorted with the most recent intervals at the top of the screen, as shown in Figure 86. The number of threads executed (DB2 accounting records per transaction/query/batch job) is shown, followed by averages per thread of elapsed and CPU times and number of SQL statements and GETPAGES.

*Average Values*

BOOLE AND BABBAGE-----		TRACE SUMMARY BY TIME		-----PERFORMANCE MGMT	
SERV ==>	TSUMT	INPUT	15:38:38	INTVL=>	3
LOG=>	N	TGT==>	DB2G		
PARM ==>	THRDHIST, INTVL=10M, SORT=ISD, GRAPH=AVG	ROW 1 OF 2	SCROLL=>	CSR	
EXPAND:	LINSEL(LTRAC), HISTORY	ENTRIES IN BUFFER	1 - 53		
11MAY00		THREAD HISTORY	11MAY00 - 11MAY00		
INTERVAL	#THREADS	AVG ELAPSED	AVG CPU	AVG #STMTS	AVG GETPGS
START	38	00:05:12	464 ms	34.9	67.3
-----	-----	-----	-----	-----	-----
19:00:00	*****	<	****	**	*****
18:50:00	****	**	*	***	*****
18:40:00	*	*	<	*****	****
18:30:00	*****	<	<	*	**
18:20:00	****	*	**	*****	*****
18:10:00	*****	*****	*****	**	*****

Figure 86. Trace Summary by Time Display with GRAPH=AVG

2. Press **PF7** to scroll down—and back in time.
3. Specify a temporary scroll value to move back to the most recent interval.

SERV ==> **M**

Press **PF8** to scroll to the top of the display.

4. To view totals instead of averages:

PARM ==> THRDHI ST, SORT=ISD, I=10M, GRAPH=TOT

Averages give you a better view of application performance, while the totals show system throughput. Note how the numbers in the column headers change, as shown in Figure 87. These are the maximum values detected, and the graphs show percent of maximum: red / highlighted if > 90 percent.

Total Values

```

BOOLE AND BABBAGE----- TRACE SUMMARY BY TIME ----- PERFORMANCE MGMT
SERV ==> TSUMT          INPUT 15:38:38 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> THRDHI ST, INTVL=10M, SORT=ISD, GRAPH=TOT      ROW 1 OF 2 SCROLL=> CSR
EXPAND: LINESEL(LTRAC), HI STORY          ENTRIES IN BUFFER 1 - 53
11MAY00                                THREAD HI STORY          11MAY00 - 11MAY00
INTERVAL      #THREADS      TOT ELAPSED      TOT CPU      TOT #STMFS      TOT GETPGS
START                38                03:17:59                17 s                616                2,560
----- |-----| |-----| |-----| |-----| |-----|
19:00:00 ***** < * ***** *****
18:50:00 **** ** ** ***** *****
18:40:00 * * * * * * * * * *
18:30:00 ***** < ***** **
18:20:00 **** * ** ***** *****
18:10:00 ***** ***** ***** ***** *****
    
```

Figure 87. Trace Summary by Time Display with GRAPH=TOT

5. Place the cursor on a line with a lot of activity and scroll it to the top with PF7.

For example, this could be a time you need to analyze because a user reported a problem.

6. Now focus on smaller time intervals by changing the INTERVAL(I) value.

PARM ==> THRDHI ST, SORT=ISD, I=01M, GRAPH=TOT

Any interval in minutes or hours is accepted, but values that divide evenly into 60 (M for minutes) or 24 (H for hours) are easier to read.

7. You can also change the sequence to sort by any column.

PARM ==> THRDHI ST, SORT=TG, I=01M, GRAPH=TOT

For example, this sorts by TOTAL GETPAGES (use the first letters of the column headers) to help identify times of high buffer activity. Or change ISD to IS to sort by INTERVAL START but with the oldest data on top.

(You could also tab to the column you want to sort on and press Enter.)

8. After the graphics display has helped you easily identify an interval to focus on, you may want to see the numbers behind the graphics.

PARM ==> THRDHI ST, SORT=TG, I=10M, GRAPH=NO

The numeric display shows both averages and totals.

This thread history is based on a standard Summary Application Trace. However, if there are other traces you want to view here, all you have to do is change the THRDHIST value in the PARM field to the traceid of any other active trace request.

## Workload Analysis

The most important benefit of providing the thread history feature through the Application Trace is that all of the trace displays are available for navigation so that analysis can be done for any workload group down to the individual threads, or up to one selective summary display, as follows:

1. Locate an interval on the display that has at least 20 threads, or the largest you can find. You may need to change the interval back to **I=1H**.

2. Use the tab key to place the cursor on that line and press **Enter**.

You are now viewing a list of each individual thread that executed in the selected interval. A **TIME** keyword has automatically been added to the parameters. This is the **LTRAC** (List Trace Entries) display that you see when selecting a trace for display (Option 4 or 5), but it is reduced to a subset of threads. (See [Figure 69 on page 70](#).)

3. Tab through the **EXPAND** line to the **PLAN** button and press **Enter**.

The panel is similar to the one you saw earlier per interval, but now the threads for the one selected interval are *summarized by PLAN*, as shown in [Figure 88](#).

*Values for Each Plan*

```

BOOLE AND BABBAGE----- TRACE SUMMARY BY PLAN ----- RX AVAILABLE
SERV ==> TSUMP           INPUT   14: 24: 47 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> THRDHIST, TIME=1810-1820, SORT=PL, GRAPH=NO      ROW 1 OF 5 SCROLL=> CSR
EXPAND: LINESEL(LTRAC), HISTORY                          ENTRIES IN DATASET 1 - 132
11MAY00
PLAN          ENTRY   AVG      AVG      AVG      AVG      TOTAL  TOTAL
              COUNT  ELAPSED  CPU     #STMTS  GETPGS  ELAPSED  CPU   #STMTS  GETPGS
-----
DSNESPSCS    2 5,516 ms  307 ms   3.0     70.0    11 s    615 ms   6     140
DSNESP RR    3 5,675 ms  446 ms  100.6   109.3   17 s    1,338 ms 302   328
DSNTIA22     12 00:01:49 281 ms  10.6   169.0 00:21:54 3,379 ms 128   2,029
DSNTIB22     11 7,330 ms  122 ms  49.0    34.0 00:01:20 1,351 ms 539   374
DSNUTIL      2 6,740 ms  186 ms   0.0    72.0    13 s    372 ms   0     144
DSN8CC22     8 6,380 ms   18 ms   7.5    10.0 00:02:20 398 ms  165   220
*** END OF SUMMARY ENTRIES ***
    
```

Figure 88. Trace Summary by Plan Name

You could also have selected one of the other **TSUM** buttons to summarize by: **AUTHID(AUTH)**, **CONNECTION(CONNECT)**, or **LOCATION(LOC)**.

**Note:** The **GRAPH=NO** change you made earlier is carried forward. You can change back to the graphic display any time by overtyping the **NO** with **YES**.

4. Place the cursor on one of the plans and press **Enter**.

Now you are back at **LTRAC** with a list of threads executed for that one plan in the interval being investigated. A **PLAN** keyword has been added to the parameters.

## Isolating Problems

You can continue this process of selecting, sorting, and categorizing until you have identified specific exception threads:

1. Select one thread (for example, one with a high elapsed time) and press **Enter**.

Now you have the complete DB2 Accounting record available to you in the STRAC (Summary Trace Entry) display. Here you can see SQL counts, buffer activity (and RID processing failures), lock counts, and elapsed/CPU times, as shown in [Figure 89](#).

*Complete  
Accounting  
Record at Your  
Finger Tips*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT  15:14:16 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> , SEQ=2                      ROW 1 OF 129 SCROLL=> CSR
EXPAND: MON(WKLD), DETAIL, CURRENT      HIST TGT--- DSN2
          ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
          SUMMARIES: SQL, SCANS, IO/LOCK, SORTS
STOP. .... 07MAY 13:35:20.20 PLAN. .... DSN8CCO TYPE. .... ALLIED
START. .... 07MAY 13:35:20.17 AUTHID. .... T127 CONNECT. .... LRTEST1/CICS
ELAPSED. .... 33 ms ORIG PRIM AUTH. .... T127 CORR ID. .... GTOOPROB
TERM. .... NORMAL/APPL END  COMMIT. .... 1 ROLLBACKS. .... 0
-----
RUNTIME ANALYSIS  IN DB2      IN APPL.      TOTAL      %IN DB2(=)      TOTAL(*)
-----
ELAPSED TIME      14 ms      19 ms      33 ms      | =====***** |
CPU TIME          12 ms      1,552 us    13 ms      | =====*      |
DB2 WAIT TIME     0 us
-----
ACTIVITY          - - - - - KEY INDICATORS - - - - -
TOTAL SQL. .... 12      SQL: SELECT= 1, FETCH= 0
GETPAGES. .... 21      SQL: DYNAMIC(PREPARE)= 3
SYNC READS (PRL=00) .... 0      I/O RSP: SYNC= 45 ms, ASYNC= 107 ms
PREFETCH PAGES READ. .... 1
UPDATES/COMMIT. .... 0.0
BFR HIT RATIOS: ... VP=100%, HP= 0%

```

Figure 89. Summary Trace Entry Display (STRAC)—Base Section

2. If you run with Accounting Class 2 or 3, don't forget to scroll down or select the **ELAPSED** button to see the graphic display of elapsed times and wait categories.
3. The other accounting detail data can be accessed by scrolling or using the expand buttons.
4. Press **PF3** to return to LTRAC, the list of completed threads.

- Tab through the EXPAND line to the **TOTALS** button and press **Enter** to access the Trace Statistics display, as shown in Figure 90.

*Qualifiers  
You've  
Selected*

```

BOOLE AND BABBAGE----- TRACE STATISTICS ----- RX AVAILABLE
SERV ==> TSTAT          INPUT 14:13:18 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> THRDHIST, PLAN=DSNTIA22, TIME=1810-1820 ROW 1 OF 37 SCROLL=> CSR
EXPAND: MON(WKLD), HISTORY
ACCOUNTING: ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLI, SPAS, DDF
SUMMARIES: SCANS, I/O/LOCK, SORTS, I/O-DB/TS
----- SUMMARY STATISTICS - ALL TRACE ENTRIES-----
----- TERMINATIONS ----- ACTIVITY-----
FIRST END..22MAY 11.07.26.38          TOTAL    AVERAGE    MAXIMUM    MINIMUM
LAST END.. 22MAY 12.04.39.68          -----
NUMBER TRANS.....191    ELAPSED 02:05:18    39 s    00:26:40    539 us
COMMIT/ROLLBK....188/....3    ELP-DB2 00:05:17    1,662 ms 00:01:13    0 us
NORMAL TERM.....189          CPU      35 s    183 ms    2,332 ms    0 us
-- NEW USER.....2          CPU-DB2  23 s    121 ms    2,079 ms    0 us
-- DEALLOC.....137    WAITS   00:04:14    1,330 ms 00:01:13    0 us
-- APPL END.....0          SQL      3,926    20        513        0
-- RESIGNON.....0    GETPAGES 17,280    90        1,857      0
-- DBAT INACT.....50    SYNC RDS  424      2         70         0
-- I FI READ.....0    PFCH PGS  846      4         252        0
ABNORMAL TERM.....1    UPD/COMI 247      1         24         0
IN DOUBT TERM.....0    BFR HIT RATIOS:..... VP= 93%, HP= 0%
----- KEY INDICATORS -----
TIMEOUTS = 5
BUFFER INCOMPLETE = 1
TOTAL DDL = 5
GRANTS / REVOKES = 7
SQL: SELECT= 209, FETCH= 2,248
SQL: INS= 155, UPD= 219, DEL= 87
SQL: DYNAMIC(PREPARE)= 247
I/O RSP: SYNC= 19 ms, ASYNC= 106 ms
LOCK SUSPENSIONS = 24
RID LIST PROCESSING USED = 314
    
```

Figure 90. Summary Statistics for All Trace Buffer Entries (TSTAT)

The TSTAT (Trace Statistics) display summarizes the data from the selected threads you were just viewing. It provides an easy way to evaluate a problem area that is broader than just one single thread execution.

The values you are looking at now summarize the performance of the one plan you previously selected. Look at the **PARM** field in the third line. You will see the selections you made by selecting lines in TSUMT (**TIME=hhmm—hhmm**) and TSUMP (**PLAN=xxxxxxx**).

- To see all activity starting at a specific time, change (or add) the **TIME** parameter with just a start time.

```
PARM ==> THRDHIST, TIME=hhmm PLAN=xxxxxxx
```

- Press **PF3** to return to LTRAC.

The parameter change is still in effect.

- To see all activity for a generic group of plans, change (or add) the **PLAN** parameter using mask characters in the plan name (plus sign for one position, asterisk for multiple positions).

```
PARM ==> THRDHIST, TIME=hhmm, PLAN=xxx*
```

The list now shows all plans starting with 'xxx'.

We have gone through examples of using **TIME** and **PLAN**, but you can also narrow your view of the workload in the same way using the other selections.

<b>A AUTHID=</b>	To select by user or user group ( <b>TSUMA</b> )
<b>C CONNECT=</b>	To select by connection name ( <b>TSUMC</b> ) ( <b>TSO</b> , <b>BATCH</b> , <b>DB2CALL</b> , <b>cicsid</b> , <b>imsid</b> )
<b>L LOC=</b>	To select by DDF location ( <b>TSUML</b> )
<b>D DAY=</b>	To select by relative day when the trace spans days (1 = date of first accounting record in the trace buffer) (can also be specified as a range of days)

Remember also that if you need to check whether there were any unusual conditions at a particular time, you can go to the Journal log (**LOG** or **PF5**) and do a **FIND** for that time to view **DB2** messages and **MAINVIEW** for **DB2** exception messages.

9. Press **PF3** to return to the Primary Option Menu.

## Review Past History

In the first dialog, you browsed the recent thread history available in the online buffers. However, this thread history trace is usually set up to log continuously to several trace log data sets, switching automatically when a log is full or when a specified switch time (for example, midnight) is reached. Here are the steps to identify whether logging is defined and to access the older data.

## Current Log Data Set

To access the current log data set:

1. From the Primary Option Menu, select the **GRAPH** option.

OPTI ON ==> **6**

As shown before, the recent thread history from the online storage buffer is displayed. (Line 4 shows ENTRIES IN BUFFER.)

2. Look at the expand buttons in line 4.

EXPAND: LI NESEL(LTRAC) , **HISTORY**

This HISTORY button tells you that logging is active and there is earlier thread history available to you.

3. Tab through the EXPAND line to the **HISTORY** button and press **Enter**.

You now have the same display of thread history, but it shows all the data on the current log data set. Depending on the way logging is set up, you may be able to see the whole day's activity, or even more. Line 4 now shows **ENTRIES IN DATASET nn - nn**.

All of the display, sort, and selection capabilities you used earlier are available here.

## Earlier Log Data Set

To access an earlier log data set:

1. If the time period in which you are interested is not in the current trace logs, you can go further back by recalling data from earlier data sets.

SERV ==> =5 (transfer to Option 5, HISTORY TRACES)

This display lists all available trace log data sets, sorted by the date and time logging was activated to each log, as shown in [Figure 91](#).

### Logged Traces

```

BOOLE AND BABBAGE ----- HISTORY TRACES ----- PERFORMANCE MGMT
COMMAND ==>>
                                                    TGT ==>> DB2G
                                                    SCROLL ==>> CSR
TIME -- 09:36:34
COMMANDS: SORT, LOCATE, NEW, STOP, START, TYPE
LC CMDS:  S (SELECT), W (SHOW), P (PRINT), D (DELETE), E (RESET)
          V (VERIFY), N (NEW), A (ARCHIVE), F (FREE)

DIRECTORY:  CIR4.LL1X.TRACEDIR
ENTRIES USED: 1,209 FREE: 201

SCROLL RIGHT >>>
LC  DATE-----TIME TRACEID  TITLE                                USERID  TGT  STAT  ACTV
   00/05/29 22:00 LEOTST01 LEOS DETAIL TRACE                   CIR4    DB2G  INV
   00/05/11 21:00 DET1    WORKLOAD DETAIL 1                   CIR2    DB2G  USED  READ
   00/05/01 00:00 THRDHIST THREAD HISTORY                       BABUSERS DB2G  UPDAT WRIT

```

Figure 91. History Traces Application

2. Locate all the data sets for the standard thread history trace.

COMMAND ==>> **SORT TR** (Sort by TRACEID)

COMMAND ==>> **L THRDHIST** (Locate the history trace logs)

All data sets created by this trace that are still cataloged are listed with the most current ones at the top.

3. Select the data set containing the time period you want to investigate.

LC (Line Command)

**S** (for Select)

The LTRAC display lists all thread entries, but you can use the expand buttons to summarize, sort, and select the data you need to see.

4. Choose **TSUMT** and then specify **GRAPH=AVG**.

This is the same display you saw for recent history in Option 6—**GRAPH**, but now it is for any time period kept online—whether a day, week, or year ago.

---

### Analyze DB2 I/O

It is also possible to start other, more detailed, system-wide traces to assist in workload analysis. However, authorization is required for such traces because of the potential overhead, so you may not be able to perform the setup for this dialog. In this case, just browse through the instructions so you understand the features offered. Then you can request such a trace when you need it from your system administrator.

In this example, you analyze I/O usage per page set (table spaces and index spaces). If you run DB2 with Accounting Class 2 or 3, increases in the average I/O wait times may indicate a problem that should be investigated. You can see this value in the ELAPSED TIME ANALYSIS section of TSTAT (select any time period or application of interest), or you may want to do a trend analysis from historical data kept in Performance Reporter tables.

## Run a System I/O Trace

To run a system I/O trace:

1. From the Primary Option Menu, select the **I/O** option.

```
OPTION ==> 7
```

The I/O Analysis Options panel is displayed, as shown in [Figure 92](#).

### Trace I/O Events

```

BOOLE AND BABBAGE----- I/O ANALYSIS OPTIONS ----- RX AVAILABLE
COMMAND ==> 7                                     TGT==> DB2G

Display Buffer Pool I/O Data:

 1 I/O BY DB/TS - Display Realtime Data by Database / Tablespace
 2 I/O BY BPOOL - Display Realtime Data by Buffer Pool
 3 I/O BY VOLUME - Display Realtime Data by Volume
 4 I/O BY DATASET - Display Realtime Data by Data Set

Display I/O Data from a Trace:

 5 CURRENT TRACES - List Current I/O Traces: n Active, nn Complete
 6 HISTORY TRACES - List All History I/O Traces

Start a DMR I/O trace:

 7 START TRACE - Review Primed Options / Activate (Start Trace Panels)

Print SMF/GTF I/O Traces

 8 DB2 TRACE - Print Instructions

```

Figure 92. I/O Analysis Options Panel

2. Go to the Start Trace request panels.

```
COMMAND ==> 7
```

You need to collect the I/O events for a short time period.

**Note:** If you do not need to access the following I/O analysis online, read the instructions in Option 8 about writing DB2 I/O trace data to SMF/GTF and printing reports.

3. The required keywords to start an I/O trace are primed in the Start DB2 Trace Request panel.

Specify any additional options necessary for this exercise and start the trace. (Possible additions to primed options are in bold type.)

```

PARM    ==> I0hhmmss           Trace ID
TYPE    ==> D                   Detail trace
STORAGE ==> 1000K               Display buffer size
LOGTRAC ==> Y                   Log to dynamically allocated data set
TITLE   ==> DB2 I/O SYSTEM TRACE Descriptive title
START   ==> hh: mm ss          Cover peak period
STOP    ==> 15                  Stop after 15 minutes
DB2AUTH ==> +                   Trace all threads
Detail Trace Options ==> *      Next panel shows I/O events selected
  SQL   ==> N                     SQL events turned off to reduce overhead
  I/O   ==> Y                     Specify I/O events
  TRBUFF ==> 50                  Increase buffers to trace many threads

```

4. Let the trace run to completion.

## Analyze I/O by Page Set

To analyze I/O by page set:

1. Return to the I/O Analysis application (Option 7 from the Primary Option Menu).

2. Browse the collected trace data.

COMMAND ==> **5** (Current Traces)

Or, if you logged the trace:

COMMAND ==> **6** (Hi story Traces)

3. Select the trace you requested (traceid IOhhmss).

LC (Li ne Command)

**S** (for Sel ect)

The first panel of trace data, a list of all threads executed in this time period, is displayed. However, go first to the summary data for the whole trace.

4. Tab through the EXPAND line to the **I/O-DBTS** button and press **Enter**.

This summarizes all I/O activity by page set for the total DB2 system. It shows total (synchronous and asynchronous) I/O measurements, as shown in [Figure 93](#).

### Summarized I/O Events

```

BOOLE AND BABBAGE----- I/O Analysis- HT ----- PERFORMANCE MGMT
SERV ==> DBIO INPUT 11:02:14 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> , SORT=DB, TOTAL LINE 1 OF 19 SCROLL=> CSR
EXPAND: PLAN, AUTH, BPOOL, TIME, CONNECT, LOCATION
EXPAND: LI NESEL (DBTS), CATALOG
OPTI ON: TOTAL, SYNC, ASYNC

11MAY00 CIR2. DB2G. JQSI0. MAY11. T1818. V01
DATA TABLE I/O I/O MAX AVG
BASE SPACE COUNT % IOWAIT IOWAIT
-----
--ms-- --ms-- 0 ... 20... 40... 60... 80
DSNDB01 DBD01 4 5.6 89 34 | ***** |
DSNDB01 DSN SCTO2 1 1.4 28 28 | ***** |
DSNDB01 DSN SPT01 1 1.4 17 17 | **** |
DSNDB01 SCTO2 2 2.8 38 31 | ***** |
DSNDB01 SPT01 4 5.6 351 95 | ***** |
    
```

Figure 93. I/O Analysis by Database / Table Space Display (DBIO)

You can see the number and percent of I/O per table space and index space, as well as the average and maximum I/O sort times. You can select only the synchronous or asynchronous (prefetch) I/Os by tabbing to the **OPTION** line. Only the synchronous I/Os directly affect application response time.

5. Tab to the **AVG IOWAIT** column and press **Enter** to sort the highest average delays to the top of the display.

The parameter is changed to **SORT=AI**. A high average I/O wait may point out possible poor DASD response times.

6. Tab to the **MAX IOWAIT** column and press **Enter** to sort the highest maximum delays to the top of the display.

The parameter is changed to **SORT=MI**. A high maximum value may point out an occasional contention problem that could be masked in the averages.

- Tab to the **I/O COUNT** column and press **Enter** to sort the highest number of I/Os to the top of the display.

The parameter is changed to `SORT=IC`. Index spaces with high I/O counts may be good candidates for a separate buffer pool configured for index data.

## Analyze I/O by Application

I/O problems may be caused by DASD response or contention problems, or they may be caused by the application just doing too many I/Os. To analyze I/O by application:

- Tab to the **PLAN** button and press **Enter** to access the I/O Analysis by Plan display, as shown in [Figure 94](#).

*I/O  
Summarized  
by Plan*

```

BOOLE AND BABBAGE----- I/O Analysis-Plan ----- PERFORMANCE MGMT
SERV ==> DBIOP          INPUT  11: 29: 20  INTVL=> 3  LOG=> N  TGT==> DB2G
PARM ==> , S0=PL, TOTAL          LINE  1 OF  3  SCROLL=> CSR
EXPAND:  PKG/PGM, LI NESEL(DBI O), CATALOG
OPTI ON:  TOTAL, SYNC, ASYNC

11MAY00          CI R2. DB2G. JQSI O. MAY11. T1818. V01
          I/O  I/O  MAX  AVG
PLAN      COUNT  %  IOWAIT IOWAIT
-----
          -----ms-- --ms-- 0 . . . 20. . . 40. . . 60. . . 80
DSNTI A31          29 40.8    89   19 | ****
DSNTI B31           6  8.5    38   25 | *****
RXDB2             36 50.7  1028   80 | *****

```

Figure 94. I/O Analysis by Plan Display (DBIOP)

This summarizes all the I/O collected by the trace by plan.

- Tab to the **I/O %** column and press **Enter**.

This sorts the applications with the highest amount of I/O to the top.

- Line select one plan.

This returns you to the DBIO display by DB/TS, but shows only the table spaces accessed by that plan.

- Try out some of the other summarizations, selection, and sorting options to narrow the focus of the displays.

Of course, all the other trace displays are available, including the Lock and I/O Summary per thread (STRAC) and the detail I/O events (DTRAC).

The next section covers printing I/O reports.

---

## Print Workload Reports

Printing workload reports is usually done from the DB2 tables of performance data supported by Performance Reporter. This gives you long-term history and trending, as well as the full flexibility of SQL for defining your own reports in addition to the predefined set. See “Reporting Facilities from DB2 Tables” in the *MAINVIEW for DB2 Performance Reporter User Guide* for more information.

Also, selective or total accounting reports, in either a short or long format, can be printed from DB2 accounting records extracted from one or more SMF files. For example, you can select from a specific time period or by plan, authorization ID, and so on. The reports also can be summarized by various criteria. See “Reporting Facilities from SMF” in the *MAINVIEW for DB2 Performance Reporter User Guide* for more information.

However, this data is often not available to answer questions about the workload until the next day. The batch trace print facility is designed to fill the need for quick reports. All the trace summary report formats are available, in any combination.

There are different types of input:

- One or more trace log data sets  
For example, of the Thread History trace.
- Archived trace logs (without reloading to VSAM)  
For example, thread history from two weeks ago.
- An SMF history file containing DB2 Accounting records or I/O trace IFCIDs
- The live SMF data sets
- A GTF trace data set

The examples in this dialog are made with the THRDHIST and system I/O traces.

To print a workload report:

1. From the Primary Option Menu, select the **HISTORY TRACES** option.

OPTION ==> 5

2. Locate an entry with a TRACEID of THRDHIST and select it for print.

LC (Line Command)  
**P** (for PRINT)

The Batch Trace Print panel where you can specify options to print one batch report is displayed, as shown in [Figure 95](#).

*Specify Options*

```

BOOLE AND BABBAGE ----- BATCH TRACE PRINT ----- PERFORMANCE MGMT
COMMAND ==>                                         TIME --- 15:00

Update job ==> N (Y/N - update job statement)      (END to edit JCL)
Title line 1 ==>
Title line 2 ==>

Data Selection:
From date ==> 10MAY2000 Time ==> 1412
To date ==> 11MAY2000 Time ==> 1125

PLAN ==>
AUTHID ==>
CONNECT ==>
CORR ==>
LOC ==>
DB2PKG ==>

REPORT SELECTION:
LTRAC ==> NO (YES/NO)
STRAC ==> NO (NO, ALL, SUMMARY, section1, section2. .)
DTRAC ==> NO (YES/NO)
POPUP ==> NO (NO/ALL/SQL/event1, event2, . . .)
TSUM ==> (A/C/L/P/T) SORT ==> (for any TSUMx)
TSTAT ==> NO (SUMMARY/ALL/NO) INTERVAL ==> 1H (for TSUM only)
DBIO ==> (X/A/C/F/L/P/T/xx) IOSORT ==> (for any DBIOx)
IOINTVL ==> 10M (for DBIOT only)
    
```

Figure 95. Batch Trace Print Panel

You can also use the sample JCL **DZTBPRNT** to tailor and submit a set of reports. This JCL is set up so that it can invoke several sample report members that explain many of the available print options.

3. If this is the first time you are using this option, you may want to update your job statements first (Update Job ==> **Y**).

- Fill in the options for a report.

REPORT ==> **traceid** Specify any name. Used as the output DD.  
DDNAME ==> Optional.  
TITLE1 ==> **any title** Optional. Centered in first report header.  
TITLE2 ==> **any title** Optional. Centered in second report header.

Data Selection is optional, but can be used to reduce the amount printed. The date and time fields are primed from the trace log data set.

REPORT SELECTION:  
LTRAC ==> N  
STRAC ==> N  
DTRAC ==> N  
POPUP ==> N  
TSUM ==> **T** SORT ==>  
TSTAT ==> **SUMMARY** INTERVAL ==> **30M**

- Press **Enter** to validate your specifications.

These options are saved in your profile and used to initialize the fields the next time this panel is requested.

- Press **PF3** to review the generated JCL in edit mode.

The options you specified are inserted into a pattern job DZJPTRAC in BBPROF. You may want to copy this JCL into your own UBBPROF data set and modify it.

- Check the other options and add if desired; for example:

TSUM=T, I=30M, **GRAPH=TOT**

Add the option to graph the totals instead of printing values. There are many other options that cannot be shown on the panel. Refer to "Printing a Trace" in Volume 2 of the *MAINVIEW for DB2 User Guide*.

- Submit the job.

COMMAND ==> **SUB**

If you prefer, you can **SAVE** or **CANCEL** the JCL.

- Press **PF3** to return to the Batch Trace Print panel.

You can make another request now if desired.

When the job completes, review the output. This gives you a graphic summary of total DB2 activity for the selected time period in 30-minute intervals, followed by a summary of the activity.

Here is another example:

```
LTRAC ==> Y  
STRAC ==> SUMMARY          (Add ,DDF if DDF is active)
```

This provides the complete DB2 Accounting information per thread. You may want to use the data selection options to limit the output.

To print an I/O report from the system I/O trace log or from an SMF/GTF data set that has DB2 I/O IFCIDs 6 - 10:

```
DBI0=X      Summarize by database/table space  
DBI0=XP     Summarize by database/table space/plan  
DBI0=PX     Summarize by plan/database/table space
```

All the other summarization options of AUTHID, connect, buffer pool, location, and time are also available.



---

## Chapter 6. Using RxD2 with MAINVIEW for DB2

In this practice session, you

1. Learn how to use RxD2/FlexTools with MAINVIEW for DB2 for SQL prototyping.
2. Access DB2 Catalog and PLAN\_TABLE information or invoke EXPLAIN directly from your terminal session for either local or remote DB2 systems (remote access is through DDF connections from a local DB2).

**Note:** This practice session takes approximately one hour to complete.

## Improve Performance with SQL Prototyping

SQL prototyping can be a valuable tuning exercise at two different times in the development cycle:

- When an application is first being developed and the basic performance characteristics of the SQL statements need to be validated against the design criteria.
- When a poorly performing SQL statement has been identified in an application and improvements are being sought.

RxD2 FlexTools provides a simple method both to execute and EXPLAIN SQL statements directly from the program source. There are often several possible methods to code an SQL statement, and the trick is to find the one with the best performance characteristics. Modifying and executing these variations while running a detail trace can simplify this task and document the results for you.

Begin by ensuring that RxD2 is accessible from your terminal session:

**Note:** You can access RxD2 only from a terminal session running under ISPF or MAINVIEW Alternate Access. Also, to run this scenario, your terminal session must be on the same MVS system as your source libraries and the DB2 system.

1. Go to the MAINVIEW for DB2 Primary Option Menu.

You will see an RX option in the middle of the panel if RxD2 is installed.

2. Request this option:

```
OPTION ==> RX          RxD2 FlexTools
```

The RxD2 Primary Option Menu is displayed, as shown in [Figure 96](#).

*RX Is Available*

```
Boole & Babbage ----- Primary Option Menu ----- RxD2 FlexTools 2.1
Option ==>                                         Target ==> DB2G
                                                    Userid ---- BOLMKW2

                                DB2 Resource Administration
-----
 1 Plans                        9 Storage Groups
 2 Packages                     10 RLF
 3 Tables                       11 DDF
 4 Table Spaces                 12 Synonyms
 5 Partitions                   13 SYSCOPY
 6 Indexes                      14 Authorization by User
 7 Index Partitions             15 Authorization by Resource
 8 Databases                    16 Stored Procedures

DB2 Application Functions          General Facilities
-----
A1 Execute SQL                    C DB2 Commands
A2 DDLGEN for Tables               D Defaults
A3 EXPLAIN PLAN_TABLE             T Tutorial
A4 EXPLAIN SQL from Edit (EXPL)   N What's New
A5 Execute SQL from Edit (TEX)    X Exit
```

Figure 96. RxD2 Primary Option Menu

3. Press **PF3** to return to MAINVIEW for DB2.

When RxD2 is installed, you will also see a short message RX AVAILABLE at the top right of many other displays.

## Start a Trace

To set up the trace for your tests:

1. Start a detail trace qualified by your AUTHID.

```
OPTION ===> ST          Start Trace
```

The default for a detail trace includes an SQL trace, which is usually sufficient information. If you also need to analyze data access by table space, you may want to add SCANS or I/O. Include LOCKS only for specific lock analysis, since this is very expensive. Log the trace so you can print reports later if desired.

If you have any questions about starting a trace request, go back through the first part of the practice session in [Chapter 4, “Tuning an Application with Trace”](#) on page 59.

2. Check that the trace has started correctly.

```
OPTION ==> CT          Current Traces
```

Press **Enter** until your trace request appears in the list and shows as active.

## Test a Statement

To start testing:

1. From the Primary Option Menu, or any command line, select RxD2.

```
OPTION ===> RX      RxD2 FlexTools
```

or

```
COMMAND ===> RX      RxD2 FlexTools
```

2. Enter ISPF/PDF Edit.

```
OPTION ===> A5      Execute SQL from Edit (TEX)
```

3. Select your program source file and member in the normal way, and then locate the statement you want to test.

(If you have the DB2 IVP installed, you can use the DSNSAMP member DSN8MCA and choose the first statement.) (Use FIND 'EXEC SQL'.)

4. Choose the statement for execution.

```
COMMAND ===> TEX
```

Place the cursor on the statement text and press **Enter**.

If you execute this from a library with an unknown source type (for example DSNSAMP), you are asked to define whether the text is in COBOL, PL/I, and so on. Otherwise, the statement is parsed immediately and primed in a panel to prepare for execution, as shown in [Figure 97](#).

### Initiate Execution

```
Boole & Babbage ----- Execute SQL -----
Command ===>                                     TARGET ===> DB2G

Location          ===> DB2G
Maximum Fetches   ===> 99999
Default SQL Action ===> COMMIT (COMMIT, ROLLBACK)
Current SQLID     ===> BOLMXW3

Enter the SQL statement below, and press ENTER to execute:
-----
SELECT CREATOR, NAME, TYPE, DBNAME, TSNAME FROM SYSIBM SYSTABLES ORDER BY CREAT
R, NAME
```

Figure 97. Execute SQL Panel

5. Determine the maximum number of FETCHes that should be executed.

```
MAXIMUM FETCHES ==> 20
```

The default is 20. If you change it here, it affects only this one execution. You also can change the default in the Defaults option (RxD2/FlexTools Primary Option Menu).

6. Define whether you want to COMMIT or ROLLBACK.

```
DEFAULT SQL ACTION ==> COMMIT
```

The default is COMMIT. In both cases, the thread is terminated at completion so that an accounting record is cut for each execution.

7. Modify the SQL text if necessary.

Review and modify the SQL statement text before execution as dynamic SQL. Host variables are not supported (they are replaced by question marks in the text), so you may have to edit in appropriate values for the test. You also can delete or add clauses. It is not necessary to edit out blanks. The text will be flowed together for execution.

8. Change the current SQLID if necessary to qualify a table name or edit it into the text.

9. Press **Enter** to submit the SQL statement for execution.

Output is returned in a scrollable display, each row separated by asterisks and the row number, followed by one line per column, as shown in [Figure 98](#).

*Results of Execution*

```

Boole & Babbage ----- Execute SQL Output ---- Row 145 to 180 of 1,174
Command ==>                                     SCROLL ==> CSR
                                                TARGET ---- DB2G

Location . . . . . : DB2G
-----
ROW # 25 ***** 5 COLUMNS
CREATOR          = BOLMXW2
NAME             = MWTTBLA
TYPE            = T
DBNAME          = MWDBTEST
TSNAME         = MUTSPCA
ROW # 26 ***** 5 COLUMNS
CREATOR          = BOLMXW2
NAME             = MWTTBL3H
TYPE            = T
DBNAME          = MWDBTEST
TSNAME         = MUTSPC3H
ROW # 27 ***** 5 COLUMNS
CREATOR          = BOLMXW2
NAME             = MWTTBL3I
TYPE            = T
DBNAME          = MWDBTEST
TSNAME         = MUTSPC3I
ROW # 28 ***** 5 COLUMNS
CREATOR          = BOLMXW2
NAME             = MWTTBL3J
TYPE            = T
DBNAME          = MWDBTEST
TSNAME         = MUTSPC3J
ROW # 29 ***** 5 COLUMNS
CREATOR          = BOLMXW2
NAME             = MWTTBL3X
TYPE            = T
DBNAME          = MWDBTEST
TSNAME         = MUTSPC3X
ROW # 30 ***** 5 COLUMNS
CREATOR          = BOLMXW2
NAME             = MWTTBL3Z
TYPE            = T
DBNAME          = DSNDB04
TSNAME         = H3TTBL1
    
```

Figure 98. Execute SQL Output Panel

10. Repeat the modification and execution of the statement for as many iterations as you need.

11. Press **PF3** to return to your edit session.

If you wish, you can now choose another SQL statement to test.

## EXPLAIN a Statement

You can EXPLAIN any statement while in edit. This can be very useful when you are first creating an SQL statement. You do not need to EXPLAIN every iteration you are testing now, since the MAINVIEW for DB2 trace captures the EXPLAIN data created for the dynamic SQL. Choose one to try out now so you know how it works:

1. Choose the statement for execution.

COMMAND ===> **EXPL**

Place the cursor on the statement text and press **Enter**.

You may be asked to define the source type, just as for TEX. Then the statement is parsed and primed in a panel to prepare for EXPLAIN, as shown in [Figure 99](#).

**Note:** You must have a PLAN\_TABLE allocated for your AUTHID or, if changed, the current SQLID.

### *Initiate EXPLAIN*

```
Boole & Babbage ----- EXPLAIN -----
Command ===>                                     TARGET ===> DB2G

Location      ===> DB2G
Catalog Prefix ===> SYSIBM
Current SQLID  ===> BOLMKW
              (Plan_Table must exist for this SQLID)

Enter the query number to use ===> 601

SQL statement to be explained:
-----
SELECT DSN8410.DEPT.DEPTNO, DEPTNAME, MGRNO, PROJNO, PROJNAME FROM DSN8410.DEPT
FULL OUTER JOIN DSN8410.PROJ ON DSN8410.DEPT.DEPTNO = DSN8410.PROJ.DEPTNO
```

Figure 99. EXPLAIN PLAN\_TABLE Qualifier Panel

2. Specify the query number to identify this iteration.

Type the query number to use ===> **1**

The default is 1.

Review / modify the SQL statement text before EXPLAINing it. Host variables are replaced with parameter markers (questions marks) in a format acceptable to EXPLAIN.

3. Press **Enter** to submit the SQL statement to be EXPLAINED.

The EXPLAIN output is returned in a formatted, scrollable display, as shown in [Figure 100](#).

**Results of  
EXPLAIN**

```

Boole & Babbage ----- EXPLAIN PLAN_TABLE ----- Row 1 to 2 of 2
Command ==>                                     SCROLL ==> CSR
                                                TARGET ---- DB2G

Catalog Prefix ==> SYSIBM
Location       ==> DB2G
PLAN_TABLE    . . . . : BOLMKW.PLAN_TABLE

Commands: SORT (Q, PL, PK, CO, DT)
LC CMDS:  P (plan detail)           T (table detail)
          PK (package detail)       X (index detail)
          M (more detail)

LC QNO  STP BLK MXSEQ PLAN      PGM      COLLECTION      DATE      TIME
-----
      601  1  1  0          RXSEL1M  RXD2              20000511  14491881
Method: First Table Accessed
Access: Sequential Scan                Prefetch: Sequential
TslOCK: IS                             Access Seq(Tabno): 1
Table: DSN8410.DEPT                   Correlation:
Index:                                  Matching  0 Keys
Sort-New Table: None                   Composite: None
-----
      601  2  1  0          RXSEL1M  RXD2              20000511  14491881
Method: Merge Scan Join                 Full Outer Join
Access: Sequential Scan                Prefetch: Sequential
TslOCK: IS                             Access Seq(Tabno): 2
Table: DSN8410.PROJ                   Correlation:
Index:                                  Matching  0 Keys
Sort-New Table: JOIN                   Composite: JOIN
-----
***** Bottom of data *****

```

Figure 100. EXPLAIN PLAN\_TABLE Output Panel

4. Most of the important information is available on this first display, but you can use line command **M** to see more detail.

5. Use the line commands to quickly access information about the DB2 objects accessed.

- LC (Line Command)
- T (for Table detail)
- X (for Index detail)

All table/index catalog information upon which DB2 bases its access path selection is easily found with these dialogs.

6. Repeat the modification and EXPLAIN of the statement for as many iterations as you need.

7. Press **PF3** to return to your edit session.

If you wish, you can now choose another SQL statement.

## Compare Test Results

To find out which of the test SQL versions ran best:

1. From edit, press **PF3** several times to return to MAINVIEW for DB2.

2. Look at your trace data.

OPTION ==> 5 HISTORY Historical Trace Data Sets

If you logged the trace, locate your data set here (SORT US, then LOCATE userid).

If not, go to Option 4, Current Traces.

3. Select the trace for viewing.

LC (Line Command)  
S (S to select the trace data)

The first display (LTRAC) lists one line for each test execution (and DB2 accounting record). You can see immediately which test executions had the best performance as measured by elapsed and CPU times, as well as number of GETPAGES.

4. Select one line for further analysis by tabbing to that line and pressing **Enter**.

You now have the complete DB2 Accounting information at your finger tips (STRAC). This provides you with statistics on wait times, locking, buffer pool activity, and I/O.

5. To compare this data to that from another iteration, use **PF10** and **PF11** to move to previous or next accounting record displays, without returning to the LTRAC list.

6. Tab through the expand buttons and press **Enter** to view summaries of SQL statistics, scans, I/O, or locks by table space (if traced).

7. To identify which SQL statement text was executed for this test iteration, tab to the **DETAIL** expand button and press **Enter**.

You are now viewing a chronological list of detailed events that occurred in DB2 while processing this statement (DTRAC display).

For dynamic SQL, this provides you with critical data to help you relate the performance data in the trace with the test SQL you executed.

8. Tab to the **BIND-TEXT** event and press **Enter** to see the complete text of the SQL statement executed, as shown in [Figure 101](#).

*Complete Text*

```

BOOLE AND BABBAGE-----  DETAIL TRACE ENTRY  -----  RX AVAILABLE
SERV ==> DTRAC          INPUT   17: 30: 58  INTVL=> 3  LOG=> N  TGT==> DB2G
PARM ==> I0111256, SEQ=1, LEVEL=2                ROW 1 OF 1  SCROLL=> CSR

START: 11: 17: 10 AUTH: BOLMKW2  PLAN: RXDB2    CORR: BOLMKW2    CONN: DB2CALL
=====
EVENT          AT          ELAPSED      CPU    DETAIL
-----
BIND-TEXT      14.892                *TYPE=DYNAMIC TEXT=SELECT * FROM SY+
=====
SELECT * FROM SYSI BM.SYSPACKAGE  WHERE NAME LI KE '%'
    
```

Figure 101. DTRAC BIND-TEXT Pop-Up Display

9. Press **PF3** to return to DTRAC.

10. Tab to the **EXPLAIN** event and press **Enter** to see the dynamic SQL EXPLAIN data, as shown in [Figure 102](#).

**EXPLAIN**  
*Data*

```

BOOLE AND BABBAGE-----  DETAIL TRACE ENTRY  -----  RX AVAILABLE
SERV ==> DTRAC          INPUT  12: 43: 58  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> MYLOCKS, SEQ=3, LEVEL=3                ROW 1 OF 18  SCROLL=> CSR
EXPAND: CATALOG
START: 12: 32: 57 AUTH: BOLSMR4  PLAN: RXDB2    CORR: BOLSMR4    CONN: DB2CALL
=====
      EVENT          AT      ELAPSED      CPU      DETAIL
-----
EXPLAIN              5. 173                *PLAN=RXDB2    COST(1. 4)
=====
QUERY NUMBER: 115          EXPLAIN DATE. .: 2000-05-11 12: 33: 0284
GROUP MEMBER: DB2H        STATEMENT TYPE: SELECT
PROGRAM NAME: RXSEL1M     COLLECTION ID. : RXD2
VERSION NAME:
WHEN_OPTIMIZ: AT BIND TIME USING DEFAULT VALUES
-----
BLKNO: 1          SEQNO: 1    MXSEQNO: 0
METHOD: FIRST TABLE
ACCESS: INDEX SCAN                PREFETCH: NONE
COL_FN:
PAGE RANGE SCAN: NO
TSLOCK: IS
TABLE: DSN8610. EMP                CORRELATION:
INDEX:                            MATCHING:
-----
BLKNO: 1          SEQNO: 1    MXSEQNO: 1
METHOD: FIRST TABLE
ACCESS: MULTIINDEX/UNION          INDEX ONLY    PREFETCH: NONE
COL_FN:
PAGE RANGE SCAN: NO
TSLOCK: IS
TABLE: DSN8610. EMP                CORRELATION:
INDEX: DSN8610. XEMP1              MATCHING: 1      KEYS
=====
    
```

Figure 102. DTRAC EXPLAIN Pop-Up Display for DB2 5.1 and Above

**EXPLAIN**  
*Data*

```

BOOLE AND BABBAGE-----  DETAIL TRACE ENTRY  -----  RX AVAILABLE
SERV ==> DTRAC          INPUT  17: 31: 16  INTVL=> 3  LOG=> N  TGT==> DB2G
PARM ==> I0111256, SEQ=1, LEVEL=2                ROW 1 OF 6  SCROLL=> CSR
EXPAND: CATALOG
START: 11: 17: 10 AUTH: BOLMKW2  PLAN: RXDB2    CORR: BOLMKW2    CONN: DB2CALL
=====
      EVENT          AT      ELAPSED      CPU      DETAIL
-----
EXPLAIN              19. 708                *PLAN=RXDB2    COST(4, 061. 4)
=====
      QUERY NUMBER  115          TIMESTAMP      1996-03-05 11: 17: 1856
      GROUP MEMBER  DB2G          COLLECTION ID  RXD2

BLK  SEQ  DESCRIPTION
-----
1    1    METHOD: FIRST TABLE
      ACCESS: SEQUENTIAL SCAN                PREFETCH: SEQ
      TSLOCK: IS
      TABLE: SYSPACKAGE                      CREATOR: SYSI BM
      INDEX:                                    CREATOR:
      SORT: NONE
    
```

Figure 103. DTRAC EXPLAIN Pop-Up Display for DB2 4.1 and 3.1

## Improve Performance with SQL Prototyping

If you need to review the catalog information, the **EXPLAIN** expand button takes you back to RxD2 to a display of the first table accessed.

11. If needed, tab to any SQL statement and press **Enter** to see the detail row statistics.

This can show you how many rows were accessed and whether the predicate was Stage 1 or 2.

12. Tab to the **LOCK SUMMARY** event to see an analysis of locking and lock states (tracing lock events is not necessary to get this).
13. As with STRAC, you can use **PF10** and **PF11** to move between the events produced by this test iteration.

## Print Results

If you logged the trace, you may now want to print a report for desk analysis or a team SQL review:

1. Press **PF3** until you are back in the History Traces display.
2. Select the print option for your log data set.

LC (Line Command)  
P (for Print)

A panel on which you can define your print options is displayed. For more details, refer to [“Print a Trace Report” on page 88](#).

Recommended print options for SQL prototyping are

LTRAC=Y	One line identifier
STRAC=SUMMARY	DB2 accounting record summary
DTRAC=Y	Detail events
POPUP=(BIND-TEXT, EXPLAIN)	SQL text and EXPLAIN data

You may want to add

NEWPAGE=TRAN	Start a new page per test execution
LEVEL=3	To see all detail events

You may also want to print the detail summaries of STRAC in a separate report (they cannot be produced in the same report with DTRAC entries).

LTRAC=Y	One line identifier
STRAC=ALL	Complete accounting with detail summaries

---

## Access DB2 Catalog and PLAN\_TABLE Information

DB2 performance is dependent on three factors:

- How the DB2 system is configured (maximum threads, buffer pool, EDM pool, logging, and so on)
- The DB2 workload (SQL optimization, lock contention, I/O patterns, and service times, and so on)
- The DB2 objects themselves (tables, table spaces, indexes, plans) and the status information stored in the DB2 catalog about them

MAINVIEW for DB2 and other performance monitors mainly present data about the first two categories. Catalog and PLAN\_TABLE access is usually a completely separate function. The ability to access RxD2/FlexTools directly from your BBI Terminal Session provides this information whenever you need it (subject to standard DB2 security).

### Accessing the RxD2 Primary Option Menu

Access to DB2 data with RxD2 is provided from any MAINVIEW product that runs in full-screen mode, not just MAINVIEW for DB2. This includes

- AutoOPERATOR and FOCAL POINT—an operator or systems programmer can check the status of DB2 resources that may be affecting availability
- MAINVIEW for CICS—a CICS systems programmer or applications manager can investigate plans and tables used from CICS
- MAINVIEW for IMS—an IMS systems programmer or DBA can investigate plans and tables used from IMS

As described in the first session, the RxD2/FlexTools Primary Option Menu is available from most MAINVIEW applications running in full-screen mode.

- From the MAINVIEW for DB2 Primary Option Menu:  
OPTI ON ===> **RX**
- From any command line:  
COMMAND ===> **RX**
- From any MAINVIEW for DB2 or MAINVIEW for IMS service display:  
SERV ===> **RX**

Now all the capabilities of RxD2 are available for use.

## Accessing EXPLAIN for Currently Executing SQL

When a DB2 application is running too long, you may want to investigate the current SQL statement being executed. The MAINVIEW for DB2 detail user (DUSER) display provides hyperlink expand buttons to perform an EXPLAIN or access existing PLAN\_TABLE EXPLAIN data.

To access EXPLAIN data for currently executing SQL:

1. Access the DUSER display:
  - a. Return to the Primary Option Menu.
  - b. Choose Option 2—ANALYZERS.
  - c. Line select **USERS** from the list of Analyzer Display Services.
  - d. Tab to an active thread and press **Enter** to access DUSER.

Check that there is an active SQL statement displayed, as shown in [Figure 104](#).

*Expand to  
EXPLAIN ==>*

```

BOOLE AND BABBAGE-----  DETAIL USER STATUS  -----  RX AVAILABLE
SERV ==> DUSER           INPUT   17: 32: 09 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> MXW3           ROW     1 OF  96 SCROLL=> CSR
EXPAND: MON(USER), UTRAC, ST(START TRACE), LOCKE, EXPLAIN, PT
      ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, SPAS, DDF
CURRENT..... 17: 32: 09. 23 PLAN..... DSNESPRR TYPE..... ALLIED
START..... 17: 22: 14. 06 AUTHI D..... MXW3 CONNECT..... TSO/TSO
ELAPSED..... 00: 09: 55 ORIG PRIM AUTH..... MXW3 CORR 1D..... MXW3
STATUS..... IN-DB2 COMMI TS..... 0 ROLLBACKS..... 0
-----
RUNTIME ANALYSIS  IN DB2      IN APPL.      TOTAL      %IN DB2(=)      TOTAL(*)
-----
ELAPSED TIME      430 ms      00: 09: 55      02: 36: 54      | <***** |
CPU TIME          64 ms      44 s           44 s           | < |
DB2 WAIT TIME     00: 09: 14      | ===== |
- - - - - ACTIVITY - - - - -      - - - - - KEY INDICATORS - - - - -
TOTAL SQL..... 2      I/O RSP: SYNC= 3,352 us, ASYNC= 12 ms
GETPAGES..... 243,322
SYNC I/O (PRLL= 0)..... 24,225
PREFETCH..... 27,644
UPDATES/COMMIT..... 0.0
BFR HIT RATIOS: ... VP= 90%, HP= 0%
-----
-SQL STATEMENT ANALYSIS - - - - -
STATEMENT #: 193      STATEMENT TYPE: OPEN
ACTIVE IN DB2      CPU TIME: 1,897 ms      ELAPSED TIME: 00: 01: 06
Package/DBRM: DSNESM68 (DYNAMIC)      PLAN ISOLATION LEVEL: RR
PROCEDURE/UDF/TRIGGER: DSN8SPAS_TEST
SELECT DISTINCT TBCREATOR, TBNAME, NAME, COLTYPE, LENGTH, SCALE, NULLS, DEFAULT
T FROM SYSIBM.SYSCOLUMNS C, SYSIBM.SYSCOLAUTH CA, SYSIBM.SYSTABAUTH TA WHERE C.T
BCREATOR = USER OR C.TBCREATOR = 'PUBLIC' OR (CA.GRANTEE = USER AND CA.TNAME = C
.TBNAME) OR (C.TBNAME = TA.TTNAME AND TA.GRANTEE = USER)
    
```

*Current SQL  
Statement ==>*

Figure 104. User Detail Status Display (DUSER)—Base Section

2. Select the **EXPLAIN** expand button.

A qualifier panel is displayed with the SQL text from DUSER.

If the DB2 target is on the same MVS as your terminal session, RxD2 accesses it directly (see the TARGET field). If the DB2 target is on a remote MVS, RxD2 accesses it through DDF. The TARGET name shown now is the local DB2 your RxD2 session is connected to (set in the Defaults panel) and LOCATION identifies the remote DB2.

3. If necessary, modify the query number and PLAN\_TABLE owner (current SQLID).

Enter the query number to use ==> **1**  
 Current SQLID ==> **your userid**

**Note:** If your SQL has unqualified table names, you may need to set up special PLAN\_TABLEs with the appropriate prefixes so that the current SQLID also can provide the proper table name qualifications.

4. Press **Enter** to invoke EXPLAIN, as shown in [Figure 105](#).

**EXPLAIN  
 Output  
 for Current  
 SQL Statement**

```
Boole & Babbage ----- EXPLAIN PLAN_TABLE ----- Row 1 to 2 of 2
Command ==>
                                SCROLL ==> CSR
                                TARGET ---- DB2G

Catalog Prefix ==> SYSIBM
Location       ==> DB2G
PLAN_TABLE . . . : BOLMXW.PLAN_TABLE

Commands: SORT (Q, PL, PK, CO, DT)
LC CMDS:  P (plan detail)          T (table detail)
          PK (package detail)     X (index detail)
          M (more detail)

LC QNO  STP  BLK  MXSEQ  PLAN      PGM      COLLECTION      DATE      TIME
-----
   601   1   1    0      RXSEL1M  RXD2          19951102  14491881
Method: First Table Accessed
Access: Sequential Scan
Tslck: IS
Table: DSN8410.DEPT
Index:
Sort-New Table: None
                                Prefetch: Sequential
                                Access Seq(Tabno): 1
                                Correlation:
                                Matching 0 Keys
                                Composite: None

-----
   601   2   1    0      RXSEL1M  RXD2          19951102  14491881
Method: Merge Scan Join
Access: Sequential Scan
Tslck: IS
Table: DSN8410.PROJ
Index:
Sort-New Table: JOIN
                                Prefetch: Sequential
                                Access Seq(Tabno): 2
                                Correlation:
                                Matching 0 Keys
                                Composite: JOIN

***** Bottom of data *****
```

Figure 105. EXPLAIN PLAN\_TABLE Output Panel

This option can be used for both static and dynamic SQL. However, for static SQL, the results may be different from the access path chosen by the DB2 Optimizer at BIND time.

5. Press **PF3** until you return to DUSER.

6. If this is a static SQL statement, select the **PT** expand button.

A qualifier panel is displayed primed with the query number (statement number) and program (DBRM or package).

7. Your user ID is primed as the PLAN\_TABLE owner.

You may need to change this specification before proceeding.

8. Press **Enter** to view the PLAN\_TABLE EXPLAIN data.

## Accessing EXPLAIN from a Trace

The previous practice session, [“Improve Performance with SQL Prototyping”](#) on page 112, showed how both SQL text and EXPLAIN information is captured in a detail trace for dynamic SQL. Neither of these events is provided by DB2 for static SQL. However, direct hyperlinks to RxD2 provide equivalent information.

To access EXPLAIN data from a trace:

1. Access a detail trace that includes static SQL:
  - a. From the Primary Option Menu, choose Option **4**—Current Traces (or Option **5**—History Traces, if you prefer).
  - b. Line select the detail trace to view the LTRAC display of traced threads.
  - c. Tab to a trace entry with several SQL statements and press **Enter** to see the accounting data for that thread (STRAC display).

- Select the SQL Summary section by tabbing to the **SQL** button in the SUMMARIES EXPAND line and pressing **Enter**.

A summary of all SQL executed by that thread is displayed, as shown in [Figure 106](#).

*SQL  
Summary*

```

BOOLE AND BABBAGE----- SUMMARY TRACE ENTRY ----- RX AVAILABLE
SERV ==> STRAC          INPUT  14: 27: 16 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> PBCR02, SEQ=5, SQL, SORT=PGM          ROW 1 OF 18 SCROLL=> CSR
EXPAND: MON(WKLD), DETAIL, HISTORY
ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRL, PKG, SPAS, DDF
SUMMARIES: SQL, SCANS, IO/LOCK, SORTS

----- SQL SUMMARY (DETAIL TRACE ONLY) -----

  STMT      AVG.      %      AVG.      %      SORT  -- PAGES SCANNED --
  TYPE      STMT    COUNT  ELAPSED  ELAP   CPU     CPU   RECS  INDX DATA WORK  REF
-----
SELECT  3228      1      25 ms   2.7   1,930 us  0.3    0     2    1    0    0
SELECT  3347      2      11 ms   2.4   4,047 us  1.5    0    11    4    0    0
OPEN    3565      6      172 us  0.1    170 us  0.2    0     0    0    0    0
FETCH   3578     11    1,986 us  2.3    782 us  1.6    0    19    3    0    0
CLOSE   3664      6      192 us  0.1    149 us  0.2    0     0    0    0    0
SELECT  3671      6    1,776 us  1.1   1,091 us  1.2    0    12    0    0    0
SELECT  3283      1    2,643 us  0.3   1,888 us  0.3    0     1    0    0    0
OPEN    3299      1     109 us  0.0     107 us  0.0    0     0    0    0    0
FETCH   3313      2     907 us  0.2    855 us  0.3    0     1    0    0    0
CLOSE   3334      1     142 us  0.0     141 us  0.0    0     0    0    0    0
OPEN    3456      1     693 ms  74.1   414 ms  74.9    8    23 1187  2    0
FETCH   3468      5     440 us  0.2    322 us  0.3    0     0    0    6    0
SELECT  4803      1    1,269 us  0.1   1,121 us  0.2    0     2    0    0    0
PGM: P025D100      44                83.9                81.0    8    71 1195  8    0

SELECT  1239      1    1,040 us  0.1   1,040 us  0.2    0     2    1    0    0
OPEN    1263      1      66 ms   7.1     47 ms  8.6   172    13    6    9    0
FETCH   1273     87     201 us  1.9    177 us  2.8    0     0    0    2    0
CLOSE   1324      1     250 us  0.0    250 us  0.0    0     0    0    0    0
OPEN    1333      1      15 ms   1.6     14 ms  2.6    7     6    4    5    0
FETCH   1343      5     254 us  0.1    254 us  0.2    0     0    0    2    0
SELECT  1375      3    1,746 us  0.6   1,165 us  0.6    0     0   12    0    0
CLOSE   1393      1     126 us  0.0    126 us  0.0    0     0    0    0    0
PGM: P025D200    100                11.4                15.0  179  21  23  18    0

SELECT  389      1      38 ms   4.2     18 ms  3.4    0    10    3    0    0
OPEN    482      1     135 us  0.0    134 us  0.0    0     0    0    0    0
FETCH   489     10     459 us  0.5    302 us  0.5    0     1    0    0    0
PGM: P025D300     12                4.7                3.9    0    11    3    0    0

** TOTALS ***      156                187    103 1221  26    0
    
```

Figure 106. SQL Summary (STRAC)

3. Select one of the static SQL statements to view the execution statistics for that SQL statement, as shown in [Figure 107](#).

*SQL  
Statistics*

```

BOOLE AND BABBAGE----- DETAIL TRACE ENTRY -----PERFORMANCE MGMT
SERV ==> STRAC          INPUT    10: 22: 51  INTVL=> 3  LOG=> N  TGT==> DB2G
PARM ==> TEST3, SEQ=000017, SQL          ROW    1 OF    18  SCROLL=> CSR
EXPAND:  SQLTEXT(EXPLAIN)

STATEMENT:  350  SELECT                                NUMBER OF EXECUTIONS:      2
PLAN:       DSNESPRR                                ELAPSED:  AVERAGE 2,794 us  TOTAL 5,588 us
PROGRAM:    DSNTIAUL                                CPU:      AVERAGE 1,271 us  TOTAL 2,543 us
LOCATION:    DB1D
PACKAGE:    SAJUYH2I                                ----- AVERAGES -----
                                                    INDEX  SEQ- DATA  SEQ- WORK
                                                    -----

      ROWS PROCESSED ALL TYPES                        4         2         0
      ROWS PROCESSED CORRECT TYPE                    4         2         0
      ROWS QUALIFIED BY DM (STAGE 1)                  3         0         0
      ROWS QUALIFIED BY RDS (STAGE 2)                 0         0         0
      ROWS INSERTED                                   0         0         0
      ROWS UPDATED                                    0         0         0
      ROWS DELETED                                    0         0         0
      PAGES SCANNED                                   8         2         0
      REFERENTIAL INTEGRITY PROCESSING:
      PAGES SCANNED                                   0         0         0
      ROWS DELETED/SET NULL                           0         0         0
    
```

Figure 107. SQL Statement Pop-Up Display (STRAC)

4. Tab to the **SQLTEXT(EXPLAIN)** expand button and press **Enter**.

**Note:** If this statement is not static SQL, the button is not highlighted. Try another statement.

This displays the complete static SQL statement text from the appropriate plan or package catalog table. From this panel, you can choose to

- Access existing EXPLAIN data in a PLAN\_TABLE  
Your user ID is primed as the PLAN\_TABLE owner. You can change it before making the request.
- EXPLAIN the text and access the RxD2 EXPLAIN display for the statement (although this may not show the access path actually used that was chosen at BIND time)

A qualifier panel is first displayed primed with the SQL text and query number 1. Your user ID is primed as the PLAN\_TABLE owner. You can change these specifications before executing the EXPLAIN. If any host variables are in the statement, they are replaced by parameter markers (question marks) to make it EXPLAINable.

- Execute the statement  
You can specify the maximum number of fetches, whether to COMMIT or ROLLBACK, and change the SQLID if necessary before execution.  
If any host variables are in the statement, they are replaced by question marks, and the text is presented on the Execute SQL panel to allow for tailoring into an executable format.

**Note:** The same panels you have viewed previously for the RxD2 EXPLAIN are shown.

5. Press **PF3** to return to STRAC.
6. Select the **DETAIL** expand button to view the detail events that occurred within the life of the thread, as shown in [Figure 108](#).

*Detail  
Events*

```

BOOLE AND BABBAGE-----  DETAIL TRACE ENTRY ----- PERFORMANCE MGMT
SERV ==> DTRAC             INPUT    10: 21: 48  INTVL=> 3  LOG=> N  TGT==> DB2G
PARM ==> TEST3, SEQ=000017, LEVEL=2          ROW    1 OF    35  SCROLL=> CSR
EXPAND: LINESEL(DETAIL), HISTORY
START: 09: 17: 11 AUTH: JEK1    PLAN: DSNTIB21 CORR: DB221REQ    CONN: BATCH
=====
EVENT          AT      ELAPSED      CPU    DETAIL
-----
CREATE-THD      0.000    45 ms 4,492 us
PLAN-ALLOC      0.045
PKG-ALLOC      0.065      ISOLATION=CS ACQ=USE REL=COMMIT
PREPARE 350    0.069 1,361 ms 26 ms *DSNESPCCS ISO=CS ACQ=USE REL=COMIT
BIND-TEXT      0.070      *RC( 0) C=DT D/X PS( 10)
EXPLAIN        0.078      *TYPE=DYNAMIC TEXT=SELECT * FROM DS+
EDM-REQ        1.302 123 ms 3,964 us *PLAN=DSN8IC22 COST(4.6)
OPEN 524      1.489 204 us 202 us DB=00000258
FETCH 532      1.489 2,058 ms 6,247 us *RC( 0) C=DT D/X PS( 2)
OPEN-TS        2.235      DB=DSN8D21A TS=DSN8S21D
OPEN-TS        3.352      DB=DSN8D21A TS=XDEPT3
FETCH 532      3.549 35 ms 441 us *RC( 0) C=DT D/X
FETCH 532      3.615 354 us 353 us *RC( 0) C=DT D/X
FETCH 532      3.616 335 us 335 us *RC( 0) C=DT D/X
FETCH 532      3.620 354 us 353 us *RC( 0) C=DT D/X
FETCH 532      3.621 386 us 387 us *RC( 0) C=DT D/X
FETCH 532      3.672 386 us 387 us *RC( 0) C=DT D/X
FETCH 532      3.673 337 us 337 us *RC( 0) C=DT D/X
    
```

Figure 108. Detail Trace Events (DTRAC)

7. Scroll down with **PF8** until you see a static SQL statement and select it to view the SQL statement pop-up display.

This looks much like the one you saw from the SQL Summary, but is only for one execution of that statement.

The same SQLTEXT(EXPLAIN) expand button is available here.

8. Press **PF3** until you return to the Primary Option Menu.

## Accessing Other Catalog Data with Direct Hyperlinks

Analyzer displays of DB2 database objects (DBATs and DBTS), as well as the detail trace dynamic SQL EXPLAIN pop-up display, provide direct hyperlinks to the related catalog information on selected objects.

To hyperlink directly to related catalog data:

1. From the Primary Option Menu, select Option 7—I/O Analysis and then select Option 1— I/O by DB/TS.
2. Line select one of the table spaces.

DBIOD—I/O Analysis by Data Set is displayed and the parameter field is primed with the name of the selected database and table space, as shown in [Figure 109](#).

*Selected Object*

```

BOOLE AND BABBAGE----- I/O Analysis-Dataset ----- RX AVAILABLE
SERV ==> DBIOD          INPUT  18:33:03 INTVL=> 3 LOG=> N TGT==> DB2G
PARM ==> TOTAL, S0=TS, DBTS=(DSN8D41A, DSN8S41E) LINE  1 OF  4 SCROLL=> CSR
EXPAND: I/O-DB/TS, I/O-BPOOL, I/O-VOL, LINESEL(DBTS), CATALOG
OPTION:  TOTAL, SYNC, ASYNC, CACHE

DATA  TABLE  DS/   I/O  I/O  MAX  AVG
BASE  SPACE   PRT   COUNT % IOWAIT IOWAIT
-----
DSN8D41A DSN8S41E 004      7 25.0   98   58 | *****
DSN8D41A DSN8S41E 003      5 17.9   36   25 | *****
DSN8D41A DSN8S41E 001      9 32.1   48   16 | ****
DSN8D41A DSN8S41E 002      7 25.0  112   45 | *****
***** END OF DATA *****
    
```

Figure 109. I/O Analysis by Data Set (DBIOD)

3. Select the **CATALOG** expand button to access a primed DB2 Table Space Administration qualifier panel; then press **Enter** to display catalog information for the selected table space, as shown in [Figure 110](#).

*Table Space Catalog Data*

```

Boole & Babbage ----- Show Table Space ----- Row 1 to 1 of 1
Command ==>
                                           SCROLL ==> PAGE
                                           TARGET ---- DB2G

Location          ==> DB2G
Catalog Prefix    ==> DB2G.SYSIBM

Commands: SORT (DB, TS, column no.)      GU (Group Utility generator)
LC CMDS:  A (authorization)                Q (generate QUIESCE JCL)
          C (generate COPY JCL)           R (generate REORG JCL)
          D (drop table space)            RC (generate REORG and COPY JCL)
          KD (generate CHECK DATA JCL)   S (show partitions)
          KX (generate CHECK INDEX JCL)   T (generate RUNSTATS JCL)
          L (list tables within)          Y (SYSCOPY recovery info)

LC
-----
DSN8S41E Status: AVAILABLE
Database:  DSN8D41A Segment Size: 0      Lock Rule: PAGE
Creator:   BOLBPL1 Page Size(K): 4       Erase Rule: N
Partitions: 4 Using: BPO                 Close Rule: N
Tables:    1 Active Pages: 120
Statstime: 1996-02-01-11.05.32.584702    Space: OKB
    
```

Figure 110. Table Space Catalog Display

From this display, you can use the line commands to browse all the other related catalog information for tables (L), partitions (S), indexes (per table), and so forth.

4. Press **PF3** until you are back at the DBIOR display.
5. Select the **CATALOG** expand button again.

Here there are no selected objects. The table space qualifier panel is presented so you can choose which table space or group of table spaces you want to view, as shown in [Figure 111](#).

*Specify Qualifiers*

```

Boole & Babbage ----- DB2 Table Space Administration -----
Command ==>                                     TARGET ==> DB2G

Location          ==> DBOG
Catalog Prefix    ==> SYSIBM
Catalog Table . . . : SYSIBM SYSTABLESPACE
Specify at least one table space qualifier.

Column           Qualifier (e.g. NULL, ^= 'AB', >123, AB++CD*)
-----
* DBNAME         ==> DSN*
  TSNAME         ==>
  CREATOR        ==>
  DBID           ---->
  OBID           ---->
  BPOOL          ==>          (Buffer Pool ID)
  PARTITIONS     ---->
  LOCKRULE       ==>          (A- any, P- page, T- table, S- tablespace, R- row)
  ERASERULE      ==>          (Y, N)
  CLOSERULE      ==>          (Y, N)
  STATUS         ==>          (A- available, C- check pending, I- incomplete)
  TABLES        ---->
  ACTIVE PAGES   ---->

                               Press ENTER to process, END to exit
    
```

Figure 111. Table Space Qualifier Panel

For example, in the DBNAME field, type **DSN\*** to display all system table spaces.

**Note:** When you are finished with this exercise, press **PF3** several times to exit the EXPLAIN function. You can initiate the next exercise from any MAINVIEW for DB2 service.

## Accessing Specific Objects in the Catalog

At other times, you may need information about a specific DB2 object or user that is being shown on a display of one of the BBI products. For example, you might be on a tabular display like LTRAC where a direct hyperlink is not available. Instead of writing the name down on a piece of paper to later type in a selection panel, the BBI—RxD2 interface provides subcommands that prompt you for immediate entry of the object name. If you decide not to type the name, you are presented with a qualifier panel that allows you to select a list of these objects.

To view details about a specific plan:

1. Type

**COMMAND** ==> **RX PL planname**

If you are in a display with a **COMMAND** line (for example, when viewing a DB2 message in the Log Display), you can type the plan name directly after **PL** on the **COMMAND** line.

If you are in a display with a **SERV** line, just type **RX PL** and press **Enter**. You are prompted at the bottom of the screen for the plan name (for example, in the STRAC trace display).

2. Press **Enter** to go to RxD2.

The **TARGET** is set to the local DB2 from your Defaults panel.

To access a different DB2 system on the same MVS as your terminal session, specify the correct **target name** on the qualifier panel. To access a remote DB2 system (if connected with DDF), keep the **TARGET** pointing to any local DB2 and specify the **location name** of the remote DB2. You also can change the catalog prefix to view an alternate catalog.

3. If you specify a plan name, RxD2 displays the DB2 Plan Administration panel with one line of information about that plan, as shown in [Figure 112](#).

*Information  
for One Plan*

```

Boole & Babbage ----- DB2 Plan Administration ----- Row 1 to 1 of 1
Command ==>                                     SCROLL ==> CSR
                                                TARGET ---- DB2G

Location      ==> NEW YORK
Catalog Prefix ==> SYSIBM

COMMANDS: SORT (column no.)
LC CMDS:  A (plan authorization)           F (free plan)
          B (bind plan functions)          P (show plan detail)
          C (list plan collections)         R (rebind plan)
          D (show plan dependencies)        S (show DBRMs and SQL statements)

          -STATUS- -
LC PLANNAME VALD OPER CREATOR  BOUND BY  DATE      TIME      I SOL VALD ACQ  REL
-----
DSNSTO1  Y    Y    BOLSMR2  BOLSMR2  950817  15353898 CS   DFER USE  COMMIT
*****
***** BOTTOM OF DATA *****
    
```

Figure 112. DB2 Plan Administration Panel

All the line commands are available to access more detailed information.

- LC (Line Command)
- A (to show plan authorizations)
- C (to list plan collections)
- D (to show plan dependencies)
- P (to show plan detail)
- S (to show DBRMs and SQL statements)

For example, you may need to see the plan dependencies or authorizations to find out why an application isn't running.

The plan detail and static SQL statement text are valuable when analyzing a detail trace for application tuning.

4. To view the EXPLAIN results for a specific plan:

COMMAND ==> **RX PT planname (owner)**

If you don't enter the operands, you are prompted at the bottom of the screen. You can specify the PLAN\_TABLE owner, or default to your user ID.

The EXPLAIN results are displayed on a panel from which catalog data about the accessed tables and indexes can be accessed directly.

LC (Line Command)  
**P** (to show plan detail)  
**T** (to show table detail)  
**X** (to show index detail)

For example, use this command when investigating a production plan that has been showing up in exception reports with increased run times. All the dependencies and statistics are available starting from this one screen, as shown in [Figure 113](#).

**EXPLAIN  
 Results  
 for One Plan**

```
Boole & Babbage ----- EXPLAIN PLAN_TABLE ----- Row 1 to 2 of 2
Command ==>
                                           SCROLL ==> CSR
                                           TARGET ---- DB2G

Catalog Prefix ==> SYSIBM
Location       ==> DB2G
PLAN_TABLE . . . : BOLMKW.PLAN_TABLE

Commands: SORT (Q, PL, PK, CO, DT)
LC CMDS:  P (plan detail)           T (table detail)
          PK (package detail)      X (index detail)
          M (more detail)

LC QNO  STP BLK MXSEQ PLAN      PGM      COLLECTION      DATE      TIME
-----
    601   1   1   0          RXSEL1M  RXD2                20000511  14491881
Method: First Table Accessed
Access: Sequential Scan                Prefetch: Sequential
Tsllock: IS                            Access Seq(Tabno): 1
Table: DSN8410.DEPT                    Correlation:
Index:                                  Matching  0 Keys
Sort-New Table: None                    Composite: None
-----
    601   2   1   0          RXSEL1M  RXD2                20000511  14491881
Method: Merge Scan Join                 Full Outer Join
Access: Sequential Scan                Prefetch: Sequential
Tsllock: IS                            Access Seq(Tabno): 2
Table: DSN8410.PROJ                    Correlation:
Index:                                  Matching  0 Keys
Sort-New Table: JOIN                    Composite: JOIN
-----
***** Bottom of data *****
```

Figure 113. EXPLAIN PLAN\_TABLE Output Panel

5. To view the details about a specific table:

COMMAND ==> **RX TB tablename | OBID**

If you don't enter the operand, you are prompted at the bottom of the screen. You can specify either the table name or OBID.

6. If you specify a table name, RxD2 displays the Show Table Objects panel, as shown in Figure 114.

*Information  
for One Table*

```
Boole & Babbage ----- Show Table Objects -----
Command ==>                                     TARGET ---- DB2G

Catalog Prefix ==> DB2G.SYSIBM
Location . . . . . :
Name . . . . . : SYSIBM.SYSTABLES
Type . . . . . : TABLE

LC CMDS: S (show detail)

LC
-----
COLUMNS          Columns In TABLE 39
-----
TABLESPACE        Tablespace: DSNDB06.SYSDBASE
                  Rows: 300      Pages: 300      TS Pct: 75%
-----
INDEXES           Indexes Defined For This TABLE 2
                  Primary Key Columns: 2
-----
RELATIONS         Referential Integrity Relationships
                  Parent: 1      CHILD:
                  Check: OK
-----
CHECKS            Check Constraints: 0
-----
DEPENDENCIES      Plans Depending On This TABLE
-----
                  Audit: NONE  Edproc:          Val dproc:
                  Statstime: 0001-01-01-00.00.000000
                  Compressed: -1%
```

Figure 114. Show Table Objects Panel

The S line command is available to access more detailed information. It provides access to related table space, column, index, key, referential constraint, or plan dependency data.

7. To view the details about a specific index:

COMMAND ==> **RX IX indexname | OBID**

If you don't enter the operand, you are prompted at the bottom of the screen. You can specify either the index name or OBID.

8. If you specify an index name, RxD2 displays the Show Index panel, as shown in [Figure 115](#).

*Information  
for One Index*

```
Boole & Babbage ----- Show Index ----- Row 1 to 1 of 1
Command ==>
                                           SCROLL ==> CSR
                                           TARGET ---- DB2G

Catalog Prefix ==> DBOG.SYSIBM
Location . . . . . : DBOG

Commands: SORT (IX, TB)                GU (Group Utility generator)
LC CMDS:  D (drop index)                C (generate CHECK INDEX JCL)
          I (show index parts)          R (generate REORG JCL)
          K (show index key columns)    T (generate RUNSTATS JCL)

LC
-----
SYSIBM.DSNDTX01      *** RUNSTATS RECOMMENDED ***      DBID: 6
Created As: Primary  On Table: SYSIBM.SYSTABLES      OBID: 61
Subpages: 1          Cluster
Bufferpool: BPO      Defined: N                      Unique First Keys: -1
Close: N             Status: N                        Unique Full Keys: -1
Key Columns: 2       Ratio: 0%                       Levels of Index: -1
Type: 2              Statstime: 0001-01-01-00.00.00.000000
Leaf Pages: -1
-----
```

Figure 115. Show Index Panel

All the line commands are available to access more detailed information.

- LC (Line Command)
- I (to show index parts)
- K (to show index key columns)

---

## Chapter 7. Printing Reports

These scenarios teach you how to print both your online and offline reports.

In this practice session, you

1. Print an online history trace and export an online windows-mode view for offline review.
2. Print offline traces from a trace log data set or SMF records.
3. Print offline Performance Reporter reports from both SMF data and DB2 tables.

This practice session takes approximately one hour to complete.

---

## Print Online Reports

You can both print online history traces and export online windows-mode views for offline review.

### History Traces (Thread Data)

In previous exercises, you learned how to run a trace and view it online. However, depending on the results, you may want to have a hardcopy for further analysis. Batch reports can also be valuable tools during an application review meeting.

**Note:** This section describes printing a trace, but you may also want to look at the accounting reports produced from SMF data, either from DB2 tables or directly from the extracted DB2 accounting records. See [“Performance Reporter Reports” on page 148](#) and the *MAINVIEW for DB2 Performance Reporter User Guide* for more information.

To begin this exercise:

1. From the Primary Option Menu, select the **HISTORY TRACES** option.

OPTION ====> 5

This panel not only provides access to view the trace data online but also offers several line commands to simplify management of the trace log data sets:

<b>W</b>	Show the options used for this trace
<b>P</b>	Generate the JCL to print a batch report
<b>D</b>	Delete this data set from the trace directory
<b>E</b>	Reset the log data set for reuse
<b>V</b>	Verify that the directory entry matches the data set contents
<b>N</b>	Add a new data set to the directory (moved from another system)
<b>A</b>	Archive the data set (only if an archive started task was specified)
<b>F</b>	Free a data set currently being read

Usually you will need only W(SHOW), D(DELETE), and P(PRINT). We are going to concentrate now on P. You can try the others when you need those functions.

**Note:** The P option only works when running under ISPF.

## All Data per Traced Thread

To print a trace report showing all data per traced thread:

1. The history traces display shows the most recent traces at the top, but you can sort on any column and use the locate command to help you find other trace logs. For example, you can sort on userid to group all your own trace logs together.

Now, select your trace log data set for print.

```
LC      (Line Command)
P      (for PRINT)
```

The Batch Trace Print panel is displayed, as shown in [Figure 116](#), where you can specify options to print a batch report.

### Printing a Trace

```

BOOLE AND BABBAGE  ----- BATCH TRACE PRINT ----- PERFORMANCE MGMT
COMMAND ==>>>                                           TIME --- 15:00

Update job ==> N (Y/N - update job statement)           (END to edit JCL)
Title line 1 ==>>
Title line 2 ==>>

Data Selection:
From date ==> 10MAY2000  Time ==> 1412
To   date ==> 11MAY2000  Time ==> 1125

PLAN      ==>>
AUTHID    ==>>
CONNECT   ==>>
CORR      ==>>
LOC       ==>>
DB2PKG    ==>>

REPORT SELECTION:
LTRAC ==> NO (YES/NO)
STRAC ==> NO (NO, ALL, SUMMARY, section1, section2. .)
DTRAC ==> NO (YES/NO)
POPUP ==> NO (NO/ALL/SQL/event1, event2, . . . .)
TSUM  ==> (A/C/L/P/T) SORT ==> (for any TSUMx)
TSTAT ==> NO (SUMMARY/ALL/NO) INTERVAL ==> 1H (for TSUM only)
DBIO  ==> (X/A/C/F/L/P/T/xx) IOSORT ==> (for any DBIOx)
                                IOINTVL ==> 10M (for DBIO only)

```

Figure 116. Batch Trace Print Panel

2. If this is your first time using this option, you must update your job statements.

```
Update Job ==> Y
```

3. Press **Enter** to display a job statement data entry panel.

Fill in the required information and return using **PF3**.

4. Set the Update Job option to **N**.

5. Fill in the options for a report.

```
TITLE1 ==> any title  Optional. Centered in first report header.
TITLE2 ==> any title  Optional. Centered in second report header.
```

Data Selection is optional. The selection fields can be used to reduce the amount printed from a long trace.

Data Selection:

From date ==> **ddmmmyyyy** Time ==> **hhmm**  
 To date ==> **ddmmmyyyy** Time ==> **hhmm**

The date and time fields are primed with the start and end date-time of the trace. You can modify them to select a shorter time period.

PLAN ==> **plan name**  
 AUTHID ==> **authorization id**  
 CONNECT ==> **connection name**  
 CORR ==> **correlation id**  
 LOC ==> **location name**  
 DB2PKG ==> **db2 package name**

The Data Selection identifiers allow you to select only a subset of threads that you need to analyze further, such as one particular plan from a Thread History (THRDHIST) trace.

REPORT SELECTION:

LTRAC ==> N  
 STRAC ==> **ALL** Print data per thread traced.  
 DTRAC ==> N  
 POPUP ==> N  
 TSUM ==>  
 TSTAT ==> N  
 DBIO ==>

There are many different reports that can be generated, either singly or combined. See [“Batch Trace Print” on page 143](#) for some examples. Browse “Printing a Trace” in Volume 2 of the *MAINVIEW for DB2 User Guide* for a full explanation of the options and to see sample reports.

The reports are based on printouts of the online displays, so the options are selected using the names of these displays, like LTRAC, STRAC, DTRAC. In this tutorial we have chosen **STRAC=ALL** as the most useful report for application tuning. It shows the following for each thread traced:

- Basic **DB2 accounting record** data
- **Environmental Indicators** section
- **Elapsed Time Analysis** section
- **SQL Statement Execution Counts** section
- **Buffer Pool Usage Analysis** section, including Global Buffer Pools
- **Lock Activity** section, including Global Locks
- **I/O Parallelism** section
- **Routines** section (stored procedures and user-defined functions)
- **DDF Summary** section (if distributed work was done)
- **Package/DBRM Overview** section (if accounting trace 7 is active)

These sections are included for detail traces that captured the relevant events:

- **SQL Summary** section with summary statistics per SQL statement (SQL events)
- **Database Summary** section with scans per page set (SCAN events)
- **Database Lock and I/O** section with locks and I/O data per page set (I/O or lock events)
- **Sort Summary** section (if any sorts were performed) (any detail trace)

6. Press **Enter** to validate your specifications.

These options are saved in your profile and used to initialize the fields the next time this panel is requested.

**Note:** Be careful! Select only the data you want to print. This report is per thread execution (like an accounting trace report from DB2PM) and can generate a large amount of output.

7. Press **PF3** to review the generated JCL in edit mode.

The options you specified are inserted into a pattern job DZJPTRAC in BBPROF. You may want to copy this JCL into your own UBBPROF profile data set and modify it.

8. Review the remaining options.

There are many more print options than can be shown on the panel, such as lines per page or maximum pages to print. The sample job contains a short description of these options. Scroll to the bottom to review them.

9. Submit the job.

COMMAND ==> **SUB**

Of course, if you prefer, you can **SAVE** the JCL for later execution, or even **CANCEL** it completely.

10. When the job is completed, review the output.

11. Press **PF3** to return to the Batch Trace Print panel. You can issue another request now if desired.

## Other Accounting Report Examples

Here are a few examples of workload-oriented reports based on the DB2 accounting record. For quick reports submitted online through the trace print panel, you will generally select one of the trace logs created by the Thread History (THRDHIST) trace for the time period you are interested in. However, these same accounting reports can be produced from any other summary or detail trace log. See [“Print Offline Reports” on page 143](#) for more complete batch reporting options.

- For a summary of total DB2 activity for the selected time period, you can first select an overview summary by time (broken into 30 minute intervals in this example), followed by a summary of all accounting data with useful averages, totals, and maximums for the complete period.

```
TSUM ==>T  
TSTAT==>SUMMARY          I N T E R V A L==>30M
```

- To change the summary to a graphic format, edit the generated selection statements in the JCL before submitting the job:

```
TSUM=T, I=30M GRAPH=TOT (or AVG)
```

- For a summary by plan (or other identifier), specify

```
TSUM=P          (or A for authorization ID, and so on)
```

All the other summarization options of AUTHID, connect, buffer pool, location, and time are also available.

- For a quick list of each thread, specify

```
LTRAC==>Y
```

- To generate a report of activity in each individual buffer pool per thread, specify

```
STRAC==>BPOOL
```

For a summary of this information per individual buffer pool for all selected threads, look at this same section in the TSTAT SUMMARY report mentioned above.

## Other Detail Trace Report Examples

You may want to try other report combinations too. Here are a few examples from detail traces:

- For an SQL statement summary per thread, sorted by Average CPU usage, followed by average SQL row processing statistics per statement:

```
LTRAC ==> Y
STRAC ==> SQL, SORTSQL=AC, SQLPOPUP
```

- For a detail event trace per thread, with pop-up displays per SQL statement:

```
LTRAC ==> Y
DTRAC ==> Y
POPUP ==> SQL
```

- For SQL statement text and EXPLAIN data for all dynamic SQL executed or BINDs of static SQL:

```
POPUP ==> (BIND- TEXT, EXPLAIN)
```

- For a summary of I/O counts and wait times per database, table space, and plan (from an I/O trace):

```
DBI O ==> XP
```

- To summarize by plan, database, and table space:

```
DBI O ==> PX
```

There are also many formatting options you can adjust:

<b>NEWPAGE</b>	Control page breaks
<b>WIDTH</b>	Specify wide (133) or narrow (81) output
<b>LINECNT</b>	Adjust the number of lines printed per page
<b>HEADING</b>	Suppress headings
<b>MAXPAGES</b>	Limit the amount of output to prevent an unexpectedly high print volume

## Online Views

You can export any windows-mode view to a data set or print it to a SYSOUT class using the **EXPort** command. The exported view can be used to supplement performance reports or can be downloaded to a workstation for use with a spreadsheet application.

When you enter the **EXPort** command, a panel is displayed requesting an existing data set name (or SYSOUT class) and formatting options, as shown in [Figure 117](#).

### *Export a View for Printing*

```

----- Export Open Data Set -----
COMMAND ==>

LIBRARY (PDS):
  Project      ==>
  Group       ==>
  Type        ==>
  Member      ==>          Replace (Y/N)? YES

Other partitioned or sequential data set:
  Data Set Name ==>
  Volume serial ==>          If not cataloged

Export Options:
  Disposition  ==> REPLACE Replace or Append if sequential data set
  Output format ==> ASIS   ASIS or CSV
  Lines/Page   ==> 0     ASIS format only (NNN)
  Sysout Class ==>      If specified, overrides other data set options

Press END to save changes and export report.
Type CANCEL to return to previous panel without saving changes.
  
```

Figure 117. View Export Panel

The output of the **EXPort** command includes all the rows of data associated with the view, even data that requires scrolling to be seen online. However, if the logical record length (LRECL) of the data set is less than the width of the view, the view data is truncated on the right.

---

## Print Offline Reports

You can print

- Offline traces from a trace log data set or SMF records
- Offline Performance Reporter reports from both SMF data and DB2 tables

## Batch Trace Print

Most scheduled batch reporting is done with the Performance Reporter reports either directly from SMF data or from that same data loaded (often summarized) in DB2 tables. However, this data is often not available to answer questions about the DB2 workload until the next day. The batch trace print facility is designed to fill the need for quick reports. All the trace summary accounting report formats are available, in any combination.

There are different types of input:

- One or more trace log data sets  
For example, the Thread History trace.
- The archived trace logs (without reloading to VSAM)  
For example, thread history from two weeks ago.
- An SMF history file containing DB2 Accounting records or I/O trace IFCIDs
- The live SMF data sets
- A GTF trace data set

A batch utility job, DZTBTRAC, is provided in your BBSAMP data set to print these reports. There is also a sample JCL member DZTBPRNT you can use when you want to generate multiple reports or print accounting reports from SMF data. The control statements for several sample reports are provided in BBSAMP member DZJPnnnn, with comments to point out some of the most useful variations.

The trace data shown in the batch reports is in the same format and content as the online displays. This includes LTRAC, STRAC, DTRAC, DTRAC pop-ups, TSTAT, and the TSUMx and DBIOx series of displays. Many of these displays can also be combined into one report. For example, a report can consist of both LTRAC and DTRAC data. Selection options, such as plan, authorization ID, or date can be used to narrow the scope of a report. In addition, you can focus in on just one area of interest, such as buffer pool activity by individual pool.

All of the report options described in [“Print Online Reports” on page 136](#) are also available by directly editing and submitting the batch trace print job, first specifying the input file and the reports to be printed.

## Trace Print from a TLDS

To print a trace log data set, use the JCL provided in BBSAMP members DZTBTRAC, as shown in [Figure 118](#). Specify the DSN of the log with the TLDS parameter. Multiple TLDSs can be concatenated.

Specify JCL  
Statements

---

```

//          JOB (ACCT), ' NAME'
//DZTBPRNT PROC TLDS=NULLFILE,          INPUT TRACE DATA SET
//          TDIR=NULLFILE,             INPUT TRACE DIRECTORY
//          SMF=NULLFILE,              INPUT UNLOADED SMF FILE
//          GTF=NULLFILE,              INPUT GTF FILE
//          ARC=NULLFILE,              INPUT ARCHIVED TRACE DATA SET
//          PFX=' HI LVL. RUN. LIB'     DSN PREFIX OF BBLINK
//PRINT    EXEC PGM=DZTBPRNT, REGION=4M, PARM=' GMWK=128K'
//*                                     INCREASE GMWK FOR LARGE TRACES
//STEPLIB  DD DISP=SHR, DSN=&PFX. . BBLINK
//SYSPRINT DD SYSOUT=*                  INPUT LIST AND DIAGNOSTICS
//SYSUDUMP DD SYSOUT=*                  ABEND DUMPS
//STD1     DD SYSOUT=*                  DEFAULT REPORT OUTPUT
//*
//*     ===>  ONLY ONE OF THE FOLLOWING INPUT DD'S MAY BE SPECIFIED
//*     ===>  IF NONE IS SPECIFIED, DEFAULT INPUT IS LIVE SMF DATASET
//*
//TRACIN01 DD DISP=SHR, DSN=&TLDS      TLDS INPUT
//TRACEDIR DD DISP=SHR, DSN=&TDIR      TLDS INPUT THRU TRACE DIR
//SMFIN    DD DISP=SHR, DSN=&SMF      SMF INPUT
//GTFIN    DD DISP=SHR, DSN=&GTF      GTF INPUT
//ARCIN    DD DISP=SHR, DSN=&ARC      ARCHIVED TLDS INPUT
//          PEND
//*
//*                                     ** SPECIFY INPUT FILE **
//PRINT    EXEC DZTBPRNT, TLDS=' SYS5. DB2P. THRDHIST. JUL01. T0001. V01'
//REPTDD1  DD SYSOUT=*                  USER-DEFINED OUTPUT DD
//RPTSUM   DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTWKLD  DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTACCL  DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTACCS  DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTSQL   DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTVNT   DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTEXPL  DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTFAIL  DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//RPTDBIO  DD SYSOUT=*                  SAMPLE REPORT OUTPUT DD
//*SYSIN   DD *                          FOR IN-STREAM REPORT STMTS
//SYSIN    DD DSN=&PFX. BBSAMP(DZJPSUM)  DEFAULT SUMMARY RPT (SUM)
//          DD DSN=&PFX. BBSAMP(DZJPWKLD) WORKLOAD SUMMARY-TIME(SUM)
//*        DD DSN=&PFX. BBSAMP(DZJPACCL)  ACCTG DETAIL- LONG (SUM)
//*        DD DSN=&PFX. BBSAMP(DZJPACCS)  ACCTG DETAIL- SHORT (SUM)
//*        DD DSN=&PFX. BBSAMP(DZJPSQL)   SQL STATEMENT SUMMARY(D-SQL)
//*        DD DSN=&PFX. BBSAMP(DZJPEVNT)  THREAD DETAIL EVENTS (D- ANY)
//*        DD DSN=&PFX. BBSAMP(DZJPEXPL)  SQL TEXT/EXPLAIN (D-SQL)
//*        DD DSN=&PFX. BBSAMP(DZJPFALL)  LOCK/EDM FAILURES (D-BASE)
//*        DD DSN=&PFX. BBSAMP(DZJPDBIO)  I/O SUMMARY BY DB/TS (D-I/O)
//*

```

---

Figure 118. JCL to Print a Trace (DZTBTRAC)

For detailed information on all the control statements in DZTBTRAC, see “Printing a Trace” in Volume 2 of the *MAINVIEW for DB2 User Guide*.

An example of the Accounting Summary Report produced when you submit this JCL is shown in Figure 119. It shows accounting totals, followed by a trace summary by plan.

BOOLE & BABBAGE	ACCOUNTING SUMMARY REPORT	PAGE: 2
REPORT: SUM		DATE: 11MAY00
		TIME: 12: 16: 56

----- SUMMARY STATISTICS - ALL TRACE ENTRIES-----				
----- TERMINATIONS -----				
FIRST END. . 01MAY 14. 37. 35. 21	TOTAL	AVERAGE	MAXIMUM	MINIMUM
LAST END. . . 01MAY 14. 38. 45. 37	-----	-----	-----	-----
NUMBER TRANS. .... 4	ELAPSED 00: 10: 20	00: 02: 35	00: 09: 12	9, 695 ms
	ELP-DB2 00: 08: 42	00: 02: 10	00: 07: 35	9, 694 ms
NORMAL TERM. .... 3	CPU 21 s	5, 302 ms	20 s	183 ms
-- NEW USER. .... 0	CPU-DB2 13 s	3, 206 ms	12 s	122 ms
-- DEALLOC. .... 3	WAITS 00: 06: 53	00: 01: 43	00: 06: 11	7, 678 ms
-- APPL END. .... 0	SQL 4, 958	1, 239	4, 954	4
-- RESIGNON. .... 0	GETPAGES 27, 541	6, 885	27, 203	56
-- DBAT INACT. .... 0	SYNC RDS 110	27	57	0
-- I FI READ. .... 0	PFCH PGS 26, 916	6, 729	26, 916	0
ABNORMAL TERM. .... 1	UPD/COMT 0	0	0	0
IN DOUBT TERM. .... 0	BFR HIT RATIOS: . . . . . VP=	2%	HP=100%	
----- KEY INDICATORS -----				
TOTAL DDL =	4			
SQL: SELECT=	0, FETCH=	4, 952		
SQL: DYNAMIC(PREPARE)=	1			
I/O RSP: SYNC=	62 ms, ASYNC=	133 ms		
LOCK SUSPENSIONS =	2			

----- ELAPSED TIME ANALYSIS (ACCTG CLASSES 2, 3 ONLY) -----				
CATEGORY	#EVENTS	AVG/EVENT	ELAPSED	%TOTAL
ELAPSED TIME				0 . . . 25 . . . 50 . . . 75 . . . 100%
IN DB2			7, 554 ms	59.01
IN APPLICATION			5, 246 ms	40.98
-- TOTALS--			13 s	100.00
WAITS IN DB2 (LOCAL)				
LOCK/LATCH	7	100 ms	703 ms	5.49
I/O WAIT	320	12 ms	3, 927 ms	30.67
LOG WRITE I/O	5	4, 083 us	20 ms	0.15
OTHER READ I/O	4	40 ms	159 ms	1.24
OTHER WRITE I/O	0	0 us	0 us	0.00
UNIT SWITCH EVENTS				
.. COMMIT/ROLLBK	4	16 ms	63 ms	0.49
.. OPEN/CLOSE	8	159 ms	1, 269 ms	9.91
.. SYSLGRNG	16	13 ms	208 ms	1.62
.. DATASPACE MGR	2	27 ms	53 ms	0.41
.. OTHER	7	7, 638 us	53 ms	0.41
ARCH. LOG(QIS)	0	0 us	0 us	0.00
ARCH. READ(TAPE)	0	0 us	0 us	0.00
DRAIN LOCK	0	0 us	0 us	0.00
CLAIM RELEASE	0	0 us	0 us	0.00
PAGELATCH CONT.	0	0 us	0 us	0.00
SPAS SERVER TCB	0	0 us	0 us	0.00
Force-at-commit	0	0 us	0 us	0.00
WAITS IN DB2 (GLOBAL)				
LOCKS	15	2, 903 us	44 ms	0.34
MSG. PROCESSING	0	0 us	0 us	0.00
---TOTAL WAITS---	383	17 ms	6, 480 ms	50.62
*NOT ACCOUNTED			289 us	0.00

----- BUFFER POOL ACTIVITY -----						
ACTIVITY	TOTAL	BP1	BP2	BP5	BP6	BP10
GETPAGES. ....	2, 545	14	592	1	6	38
SYNC READS. ....	320	11	0	0	0	0
GETPAGES/READIO. ....	8.0	1.3	0.0	0.0	0.0	0.0
COND. GP FAILURES..	0	0	0	0	0	0
SEQ. PREFETCH REQ.	22	2	0	0	0	6
LIST PREFETCH REQ.	1	0	0	0	0	0
DYNAMIC PREFETCHES.	16	1	0	0	0	0
ASYNC PAGES READ...	284	47	0	0	0	0
PAGES/PREFETCH REQ.	7.3	15.7	0.0	0.0	0.0	0.0

BOOLE & BABBAGE	ACCOUNTING SUMMARY REPORT	PAGE: 4
REPORT: SUM		DATE: 11MAY00
		TIME: 12: 16: 56

I/O SYSTEM TRACE									
PLAN	ENTRY	AVG	AVG	AVG	AVG	TOTAL	TOTAL	TOTAL	TOTAL
	COUNT	ELAPSED	CPU	#STMTS	GETPGS	ELAPSED	CPU	#STMTS	GETPGS
DSNTIA41	11	483 ms	172 ms	11.3	162.3	5, 309 ms	1, 889 ms	124	1, 785
DSNTIB41	1	18 s	5, 954 ms	1.0	20.0	18 s	5, 954 ms	1	20
DSN8CCO	2	00: 01: 07	19 ms	12.0	7.0	00: 02: 13	38 ms	24	14
RXDB2	1	6, 086 ms	2, 633 ms	245.0	1, 160	6, 086 ms	2, 633 ms	245	1, 160
***** END OF SUMMARY ENTRIES *****									

Figure 119. Accounting Summary Report

All control of the input and formatting is done through a series of keywords in the input job stream (or in a PDS member) under ddname SYSIN. Keywords that apply to all requested reports can be specified following a label of GLOBAL. Keywords that apply to a specific report must be specified following the REPORT label. An example of two report requests is shown in [Figure 120](#).

**Specify  
Request  
Keywords**

---

```
GLOBAL  TIME=1300- 1500
*
*      BIND-TEXT and EXPLAIN FROM TSO
*
REPORT  REPORTID=DAI LY1, LTRAC=YES, NEWPAGE=TRAN,
        POPUP=(BI ND- TEXT, EXPLAI N, OPEN) ,
        CONNECT=TSO,
        TITLE1=' DETAILS OF BI ND AND EXPLAI N' ,
        TI TLE2=' WITH STATS FROM OPEN' ,
        DDN=OUTPUT1, WI DTH=WI DE          COMMENTS ABOUT THE REPORT
*
*      FULL EXAMI NATION OF CERTAIN SPECIAL TRANSACTIONS
*
REPORT  REPORTI D=SPECI ALS, LTRAC=YES, STRAC=SUMMARY, DTRAC=YES, POPUP=ALL,
        NEWPAGE=(TRAN, FI RSTEVENT) , LEVEL=3,
        PLAN=(PAY+++++, ACCT1+++), CONNECT=I MSP,
        AUTHI D=(USR1, USR5)
```

---

Figure 120. Sample Report Requests

The GLOBAL keyword TIME limits the trace output to 1pm to 3pm for both reports.

The first report prints the one-line LTRAC entry on a new page for each transaction of the selected TSO threads and the pop-ups for BIND-TEXT, EXPLAIN, and OPEN after that one-line entry for each transaction. The report is written on ddname OUTPUT1.

The second report prints the one-line LTRAC entry, the STRAC accounting summary, and all DTRAC events and pop-ups for selected plans from two specific IMS users. The report is written on ddname SPECIALS.

For detailed information about each of the request keywords you can use, see “Printing a Trace” in Volume 2 of the *MAINVIEW for DB2 User Guide*.

## Trace Print from SMF Records

This same batch utility (DZTBTRAC), shown in [Figure 118 on page 144](#), can be used to print summary trace reports from DB2 accounting records written to GTF or SMF, either from an SMF history tape or from the live SMF data sets.

In addition, the following I/O analysis reports can be produced from an SMF or GTF file that includes the I/O trace records (IFCIDs 06-10):

DBIO=X

I/O analysis by database and table space

**Note:** This report can be summarized by many different summary key combinations, such as plan, database, table space or database, table space, plan. See “Special Report Summarization Options for DBIO” in Volume 2 of the *MAINVIEW for DB2 User Guide* for examples.

DBIO=A

I/O analysis by AUTHID

DBIO=C

I/O analysis by connection name

DBIO=F

I/O analysis by buffer pool

DBIO=L

I/O analysis by location

DBIO=P

I/O analysis by plan

DBIO=T

I/O analysis by time interval

**Note:** Other detail trace events are not extracted from SMF or GTF. In addition, you cannot process SMF or GTF files in the same run with TLDSs.

### Performance Reporter Reports

Performance Reporter is an offline analysis system that produces reports that can be used to evaluate DB2 system and application performance. These evaluations can be used for DB2 planning, forecasting, and performance management.

Several reports can be produced from SMF extract files without loading the data into DB2 tables. This includes accounting short and long reports and statistics short and long reports.

However, the performance data tables are the main source of historical information for Performance Reporter. You can print any number of reports produced from the summary and detailed accounting tables, or statistics, buffer statistics per pool, and audit tables.

### Reports from SMF

Selective or total accounting reports, in either a short or a long format, can be printed from DB2 accounting records extracted from one or more SMF files. For example, you can select from a specific time period or by plan, authorization ID, and so on. The reports also can be summarized by various criteria.

Use the DPREPORT batch job to print accounting or statistics reports, in either a short or a long format. [Figure 121 on page 149](#) shows sample job control statements for producing an accounting long report ordered by primary authorization ID and plan name, summarized in 8-hour intervals.

*Specify Job  
Control  
Statements*

---

```
//DPRACCT JOB
//*-----
//*          JOB TO PRODUCE AN ACCOUNTING LONG REPORT          -
//*          ORDERED BY PRIMARY AUTHID AND PLAN NAME          -
//*          FROM 00:00:00 - 23:59:59 HOURS FOR OCT. 16 - 18,  -
//*          SUMMARIZED IN INTERVALS OF 8 HOURS (480 MINUTES). -
//*          ONLY AUTHIDS OF BPL2X OR THOSE BEGINNING WITH CJN* -
//*          ARE SELECTED FOR THIS REPORT.                      -
//*-----
//STEP1 EXEC PGM=DPREPORT, REGION=4096K
//STEPLIB DD DSN=HI LVL. DPRLOAD, DISP=SHR
//DPDACCT DD DSN=HI LVL. ACCT31, DISP=SHR
//SYSPRINT DD SYSOUT=*
//DPDPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
//SYSUDUMP DD SYSOUT=*
//SORTWK01 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK02 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK03 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK04 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK05 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK06 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK07 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SORTWK08 DD UNIT=SYSDA, SPACE=(TRK, (45, 15)), DISP=(, DELETE, DELETE)
//SYSIN DD *
REPORT=ACCT
REPTYP=LONG
RSMFID=SYSB
RDB2ID=DB2F
FRDATE=20000516
FRTIME=000000
TODATE=20000518
TOTIME=240000
INTVAL=480
RORDER=(PRAUTH, PLANAM)
FILTR1=(BPL2X, CJN*)
//
```

---

Figure 121. Sample DPREPORT Job Control Statements for Accounting Reports

See “SMF Reporting Facilities (DPREPORT)” in the *MAINVIEW for DB2 Performance Reporter User Guide* for detailed information about each of the JCL control statements in the DPREPORT job.

### Reports from DB2 Tables

Printing workload reports is usually done from the DB2 tables of performance data supported by Performance Reporter. This gives you long-term history and trending, as well as the full flexibility of SQL for defining your own reports in addition to the predefined set. See “DB2 Tables Reporting Facilities” in the *MAINVIEW for DB2 Performance Reporter User Guide* for more information.

Performance Reporter provides predefined reports using SQL statements, which can be run through a batch reporting facility (DPRREPT) or through QMF queries. DB2 performance charts and plots are also provided, which are available through QMF only. Other queries and reports based on the performance data tables can also be defined.

#### Predefined Reports

The prepared reports included with Performance Reporter show DB2 system workload to help the DB2 performance analyst, capacity planner, or service manager solve specific DB2 problems. These reports are generated from accounting, statistics, and audit data.

With these reports, standard reporting can be run on a daily or weekly basis or both.

**Daily run**            The queries processed by the batch reporting program, DPRREPT, from the detail statistics (queries STxxx) and detail accounting (queries ACxxxx) tables are set up to produce a set of daily reports on the data loaded from the previous day (current date minus one). The sample JCL in BBSAMP named DPRRPT includes all distributed reports. Run these reports once, select those ACxxxx and STxxx reports you want to review daily, and create a job for this daily run.

**Note:**    If you do not load detail accounting records into table DMRACDTL, but only data summarized by hour or day into table DMRACSUM, delete the ACxxxx reports from the job and modify the date selection on the SAxxxx reports to produce your daily reports. The report results will be the same. Only the detailed exceptions report cannot be produced.

**Weekly run**        The queries processed by the batch reporting program, DPRREPT, from the summary accounting table (queries SAxxxx) are set up to produce a set of weekly reports on the data from the previous week (current date to current date minus seven). Select the SAxxx reports you want to review weekly and create a job for this weekly run.

Any of the distributed SQL queries can be modified or used as a model to produce queries to satisfy ad hoc reporting needs. However, the flexibility of QMF, in general, makes it the better vehicle for such reporting. An added advantage with QMF is that the procs provide an easy way to select a specific time period for reporting without modifying the queries themselves.

## QMF Queries

QMF can be used to run distributed reports or tailor custom performance reports and charts. A generalized QMF procedure, DZPRQRPT, runs predefined Performance Reporter queries and formats the reports using the distributed Performance Reporter forms. The predefined reports produced with QMF are the same as those produced with the batch reporting facility, DPRREPT. However, with QMF, you can select a range of dates for these reports.

**Note:** You must enter the QMF program to use this procedure. Detailed instructions to run and use QMF can be found in the *IBM Query Management Facility Learner's Guide*.

To run the predefined Performance Reporter queries from QMF, type the command:

```
RUN DMRPR. DZPRQRPT (&REPORT=report name
```

where `report name` is the name of the report you want to produce. For example, to run the DB2 Accounting Overview Report, type

```
RUN DMRPR. DZPRQRPT (&REPORT=ACOVERA
```

The report table date range prompts FROM and TO might appear when running these queries under QMF. Type the date range you want using the standard TSI format:

```
' YYYY- MM- DD- HH. MM. SS. TTTTTT'
```

Or, you can set global report variables by typing one of these procs:

```
RUN DMRPR. DZPRQDAT
```

```
RUN DMRPR. DZPRQDEF
```

Once you have set these global variables, they will be used by all Performance Reporter report queries for the duration of the QMF session.

To produce QMF batch reports, use the sample jobs in BBSAMP member DZPRQBAT.

### Customizing Your Own Reports

Once you have used the Performance Reporter facilities to load data into DB2 tables, you may want to use your own reporting tools to create customized reports.

However, since Performance Reporter uses either QMF or a generalized report generator to provide reporting and charting facilities, the entire report is defined by the SQL select statement used to read the data tables. Thus, it is very easy to create customized reports for a specific requirement. The SQL select statements used to produce the distributed reports and charts are available, within the QMF or DPRREPT environments, as models in tailoring new reports or charts.

The JXREPT program is designed to run any SQL SELECT statement against the Performance Reporter tables and report the results.

See “Customizing Reports” in the *MAINVIEW for DB2 Performance Reporter User Guide* for more information.

One example of a custom report you may want to create is to change the time interval for reporting statistics data.

The default statistics reports show statistics by the DATETIME stamp in each record. To provide a historical view, you may want to modify the distributed reports to show statistics grouped by a longer time interval. To simplify this type of reporting, the statistics tables include columns for several other date and time values, such as DATE, MONTH, DAY, TIME, and HOUR.

**Note:** Data is not *spread* across intervals.

An example of a report by DATE and HOUR is in BBPARM member STOVERH, as shown in [Figure 122](#). See “Reporting Statistics Data by Time Interval” in the *MAINVIEW for DB2 Performance Reporter User Guide*.

BOOLE AND BABBAGE		DB2 STATISTICS OVERVIEW REPORT - AVERAGES										PAGE 0002			
REPORT: STOVERH												REPORT DATE: 2000-05-25 11.14.54			
DATE FROM: 2000-05-01 00:09:01												LOCATION - SANJOSE			
DATE TO: 2000-05-19 23:55:49												SUBSYSTEM - DB2P			
DATE / HOUR	CREATE	PHASE 2	SYNC		SQL	SQL	SQL	GETPAGES	PAGE	READ	WRITE	EDM	LOG	WRT	LOCK
	THREADS	COMMITTS	COMMITTS	ABORTS	MANIP.	CONTROL	DEFINIT	REQS	UPDATES	I/O	I/O	LOADS	CALLS	SUSPEND	
2000-05-01 00	1	0.0	0.0	0.0	0.0	0.0	0.0	6.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
2000-05-01 01	0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2000-05-01 02	8	0.6	0.5	0.1	1.2	2.2	0.0	367.7	176.7	15.7	1.3	1.3	4.0	0.0	0.0
2000-05-01 03	3	0.3	1.0	0.0	0.6	0.0	0.0	64.6	30.6	1.6	4.0	0.0	2.0	0.0	0.0
2000-05-01 04	1	0.0	1.0	0.0	253.0	0.0	0.0	520.0	0.0	15.0	0.0	2.0	0.0	0.0	0.0
2000-05-01 05	1	2.0	0.0	0.0	1.0	0.0	0.0	10.0	0.0	8.0	0.0	0.0	0.0	0.0	0.0
2000-05-01 06	14	1.0	0.0	0.0	1.0	0.8	0.0	17.3	0.0	4.8	0.0	0.2	0.0	0.0	0.0
2000-05-01 07	10	0.6	0.7	0.0	1.4	1.5	0.0	304.6	168.1	4.7	2.4	0.0	3.4	0.0	0.0

Figure 122. Statistics Overview Report by DATE and HOUR

---

# Index

## Symbols

- # prefix
  - definition 27
- #SQLM monitor
  - tutorial example 27
- \* (asterisk)
  - wild card
    - tutorial example 98
- + (plus)
  - qualifier
    - tutorial example 98
- @ prefix
  - definition 27

## A

- accessing
  - MAINVIEW 6
- accounting records
  - tutorial example
    - LTRAC display 71
    - STRAC display 72, 97
    - TSTAT display 85
- active requests
  - list of monitors 26, 32
- active threads
  - all 11
  - multiple DB2s 13
- Active Timer Request list
  - tutorial example 26, 32
  - select by area 33
- Activity Overview view 8, 15
- Activity Rates view 9
- Add SSI Context Definition panel 43
- All Active Threads view 11
- analyzer
  - tutorial example
    - Primary Option Menu selection 7
- application I/O analysis
  - tutorial example
    - activity by package/program 86
    - activity by page set 86, 104
    - activity by plan 86, 105
    - activity by SQL statement 87
    - asynchronous I/Os 86, 104
    - collecting I/O events 103
    - detail I/O events 87, 103
    - printing I/O reports 106
    - running a system I/O trace 103
    - summarized trace events 86, 87, 104
    - synchronous I/Os 86, 104
    - system-wide traces 102
    - total measurements 104
- application trace
  - tutorial example
    - access services 70
    - accessing EXPLAIN 124
    - activating 66
    - additional options 62
    - analyzing DB2 workload 93
    - avoiding expensive lock event traces 81
    - BIND-TEXT pop-up 118
    - buffer pool activity 74
    - collecting I/O events 103
    - database summary 79
    - defaults 61
    - detail entry display (DTRAC) 83, 118, 127
    - detail event summaries 76
    - detail events 64, 103
    - elapsed time 73
    - error messages 66
    - exception filters 63
    - EXPLAIN pop-up 119
    - I/O analysis 86, 102
    - improve performance 67, 112
    - list of current 60, 69
    - list of historical data sets 69, 88, 101, 136
    - list of trace entries (LTRAC) 71
    - lock and I/O events 80
    - LOCK-SUMMARY pop-up 119
    - logging 65
    - long-running threads 69
    - managing log data sets 88, 136
    - narrowing 63
    - navigating 70, 82
    - package/DBRM overview 75
    - printing 88, 106, 136
    - qualifying 62
    - sort summary 81
    - SQL statement pop-up 78, 84, 126
    - SQL statement summary 76, 82, 125
    - starting 60, 103, 113
    - summary by authorization ID 99
    - summary by connection name 99
    - summary by DDF location 99
    - summary by plan 96
    - summary by relative day 99
    - summary by time 94, 100
    - summary entry display (STRAC) 72, 97
    - summary of all threads (TSTAT) 85, 98
    - test applications 67, 112
    - tuning 59, 69
- application transfer
  - tutorial examples 28, 30, 101
    - CT command 113
    - LOG command 38, 99
    - SM command 26
    - ST command 60, 113
    - to full-screen mode 16

- application tuning
  - with trace
    - tutorial example 59, 69
- AREA command
  - tutorial example 27, 33
- asterisk (\*)
  - qualifier
    - tutorial example 98
- ASYNCR parameter
  - tutorial example 86
- ATTN key
  - terminating GO
    - tutorial example 34
- audit trail 38
- authorization
  - for tutorial exercises
    - cancel a thread 12
    - issue DB2 commands 40
    - issue RESET USERID command 59
    - modify or purge a monitor 29
    - start an application trace 59, 60, 102
    - update BBPARM data set 59
- automatic multiple requests startup 26, 32
- automatic refresh mode
  - monitor history display
    - tutorial example 34
- automatic warning 26, 28, 32
- AutoOPERATOR
  - triggering EXECs 28
- average data measurement 27

## B

- balancing
  - table space allocations
    - tutorial example 25
- batch trace print
  - tutorial example 88, 106, 136, 143
    - data entry panel 89, 107, 137
    - SQL prototype 120
- BFRPL view
  - tutorial example 17
- BFRPLD view
  - tutorial example 17
- BFRPLHD view
  - tutorial example 18
- bind
  - text pop-up
    - tutorial example 84, 118
- BLKDMRW
  - default set of monitors 26
- blocking requests
  - automatic start 26
- buffer pool
  - tutorial example
    - activity summary (STRAC) 74
    - all defined pools 17
    - hit ratios 18
    - one pool 17
    - tune for data sharing group 54

- Buffer Pool Page Set GBP-DEP Summary view 48
- Buffer Pool Page Set Summary - Session view 25
- Buffer Pool Rates Detail view 18
- Buffer Pool Statistics - SSI view 17

## C

- cache
  - statistics
    - tutorial example 24
- cancel thread
  - from DUSER display 12
- card
  - quick reference 1
- CATALOG hyperlink 128, 129
- check status and activity 7, 15
- codes
  - severity 31
- commands
  - application transfer
    - tutorial examples 26, 28, 30
    - windows to full-screen mode 16
  - DB2
    - displaying response 40
    - issuing from MAINVIEW for DB2 40
- COMMIT
  - execute SQL 114, 126
- compare threads 82
- CONACT view
  - tutorial example 42
- CONASEL view
  - tutorial example 42
- console
  - MVS
    - warning messages sent to 28
- contention
  - lock
    - tutorial example 19, 32
    - with data sharing 50
- context
  - defining
    - for a data sharing group 42
    - selecting 42
- CONTEXT command
  - tutorial example 14, 42
- count data measurement 27
- critical problems
  - review
    - tutorial example 30
- CT application transfer
  - tutorial example 113
- current
  - hit ratios 18
  - SQL statement
    - full text 12, 122
  - SQLID 115, 122, 126
  - thread activity 11
    - multiple DB2s 13
- Current Traces application
  - tutorial example 60, 69

## D

- Data Collection Monitor list application
    - tutorial example
      - scroll 27
      - select by area 27, 33
      - sort 27
  - data selection identifiers
    - tutorial example 138
  - data sets
    - tutorial examples
      - I/O analysis 128
  - data sharing
    - buffer pool tuning 54
    - current group activity 45
    - defining the group context 42
    - lock contention 50
    - monitoring a group 41
    - page set considerations 46
  - Data Sharing Wizard GBP Directory Detail view 56
  - Data Sharing Wizard GBP Directory Entries view 56
  - Data Sharing Wizard GBP Options view 55
  - Data Sharing Wizard GBP Size Activity view 57
  - Data Sharing Wizard GBP-DEP/Member view 47
  - Data Sharing Wizard Group Members view 54
  - DB2
    - accounting record
      - LTRAC display 71
      - STRAC display 72, 97
    - commands
      - displaying response 40
      - issue from MAINVIEW for DB2 40
    - Exceptions display
      - tutorial example 31
    - messages
      - logging 38
      - selecting current target DB2 39
      - viewing a subset 39
    - Performance Management
      - isolating problems 5, 97
    - status 7, 15
      - detail view 15
  - DB2AUTH keyword
    - tutorial example 62, 103
  - DB2CONN keyword
    - tutorial example 28
  - DB2EX display
    - tutorial example 31
  - DB2PLAN keyword
    - tutorial example 62
  - DB2TRACE authorization parameter
    - for tutorial exercises 59
  - DBIO display
    - tutorial example 86, 104
  - DBIOD display
    - tutorial example 128
  - DBIOP display
    - tutorial example 105
  - DBIOS display
    - tutorial example 87
- DBRM
    - overview (STRAC)
      - tutorial example 75
  - defaults
    - for starting a monitor 27
    - for starting a trace 61
    - query number 116
    - starter set of monitors 26
  - definitions
    - all SSI contexts 42
  - detail trace
    - tutorial example
      - avoiding expensive lock event traces 81
      - BIND-TEXT pop-up 118
      - database summary 79
      - detail entry display (DTRAC) 83, 127
      - display level 83
      - event summaries (STRAC) 76
      - EXPLAIN pop-up 119
      - I/O analysis 87, 104, 105
      - lock and I/O events 80
      - LOCK-SUMMARY pop-up 119
      - navigating 82
      - sort summary 81
      - SQL statement pop-up 78, 84, 126
      - SQL statement summary 76, 82, 125
      - starting 64, 103
      - testing SQL 118
  - detail user display
    - tutorial example 11, 122
  - directory
    - tutorial example
      - too few entries 55
  - DMAREAZ view
    - tutorial example 36
  - DMRBEX00
    - logging DB2 messages
      - tutorial example 38
  - DMWARN view
    - tutorial example 35
  - DOBJ view
    - tutorial example 37
  - DTRAC
    - display
      - avoiding expensive lock event traces 81
      - BIND-TEXT pop-up 118
      - EXPLAIN pop-up 119
      - LOCK-SUMMARY pop-up 119
      - SQL statement pop-up 84
      - testing SQL 118
      - tutorial example 83, 127
  - DUSER display
    - tutorial example 11, 122
  - DZJPTRAC skeleton JCL member
    - tutorial example
      - printing a trace 90, 108, 139
  - DZTBPRNT program 107, 143
  - DZTBTRAC utility 143

## E

### Easy Menu

- EZDB2 (one DB2)
  - tutorial example 14
- EZDSSI (multiple DB2s)
  - tutorial example 8, 35

- elapsed time analysis
  - tutorial example 73

### Enhanced Journal Facility 39

#### error messages

- current traces
  - tutorial example 66

#### event pop-up displays

- tutorial example
  - BIND-TEXT 84, 118
  - EXPLAIN 84, 119
  - LOCK-SUMMARY 84, 119
  - LOCK-TMO 84
  - MULTI-INDEX 84
  - SQL statement 84, 119

#### exception

- conditions
  - system 13
  - user 13
  - warning monitors 13
- locate time
  - tutorial example 38

### Exception Conditions view 13

#### Exception Filters

- tutorial example 63

#### exception messages

- displaying 31

### EXEC

- application transfer command
  - tutorial example 30
- triggered when threshold is exceeded 28

#### execute SQL

- tutorial example 114
  - from a trace 126
  - results of execution 115

#### expand

- tutorial examples
  - BIND-TEXT pop-up 118
  - from DBIOD display 128
  - from DBIOR display 128, 129
  - from DTRAC display 118, 119
  - from DUSER display 122, 123
  - from LTRAC display 71, 85, 86
  - from SQL statement pop-up 78, 126
  - from TSTAT display 86
  - from USERS display 122
  - to a subset of threads (LTRAC) 96
  - to buffer pool activity 74
  - to current trace log data set 100
  - to database summary 79
  - to DB2 catalog information 128
  - to DBIO display 86, 104, 105
  - to DBIOD display 128
  - to DBIOP display 105

#### expand (*continued*)

- tutorial examples (*continued*)
  - to DBIOS display 87
  - to detail entry display (DTRAC) 83, 127
  - to DTRAC display 118
  - to DUSER display 122
  - to elapsed time analysis 73
  - to EXPLAIN PLAN\_TABLE display 122, 123
  - to EXPLAIN pop-up 119
  - to lock and I/O events 80
  - to LOCK-SUMMARY pop-up 119
  - to package details 75
  - to sort summary 81
  - to SQL statement pop-up 78, 84, 126
  - to SQL statement summary 76, 82, 125
  - to SQL statement text 78, 126
  - to STRAC display 71, 97
  - to TSTAT display 85, 98
  - to TSUMP display 96

#### EXPLAIN from edit 116

#### EXPLAIN hyperlink

- from a trace
  - tutorial example 124

#### EXPLAIN PLAN\_TABLE

- output panel 117, 123
- qualifier panel 116, 122
- tutorial example
  - accessing from a trace 126
  - accessing more details 132
  - information for one plan 132

#### EXPLAIN pop-up

- tutorial example 84, 119

#### EXPort command 142

## F

### Fast Menu 16

#### fetches

- execute SQL
  - tutorial example 114, 126

#### filtering data

- tutorial examples 23

#### filters

- tutorial example 63

#### FIND command

- tutorial example 38

#### format

- options
  - batch trace print 91, 141

## G

- global
  - lock contention 50
- Global Lock Wizard view 50
- Global Lockout Events view 52
- GO command
  - tutorial example 34
- goals
  - response time
    - monitoring 37
- graph
  - tutorial example
    - average values 94, 101
    - CPU time 72
    - elapsed time 72, 73
    - I/O wait time 86, 87
    - monitor history display 33, 36
    - package/DBRM overview 75
    - Primary Option Menu selection 94, 100
    - total values 95
    - trace summary by time 94, 100
    - warning status of active timer requests 26, 32
    - workload objectives 37
- group buffer pool
  - dependent page sets 47
  - tuning 54
- GROUPSQL keyword
  - tutorial example 64

## H

- headers
  - suppress 141
- help
  - commands 23
  - tutorial example 29, 48
- history
  - expand button
    - tutorial example 100
  - reviewing recent
    - tutorial example 15
  - short-term 26
  - trace directory
    - tutorial example 69, 88, 101, 136
  - workload
    - analyze recent 94
    - review past 100
- hit ratios 18
- host variable support 115, 116, 126
- hyperlink
  - CATALOG
    - tutorial example 128, 129
  - EXPLAIN 122, 124
  - PT 123
  - SQLTEXT(EXPLAIN) 78, 126
  - to RxD2
    - from a trace 124
    - from DBIOD 128
    - from DBIOR 129
    - from DUSER 122, 123
    - from SQL statement pop-up 78, 126
    - to catalog data 128, 129

## I

- I/O analysis
  - tutorial example
    - activity by package/program 86
    - activity by page set 86, 104
    - activity by plan 86, 105
    - activity by SQL statement 87
    - asynchronous I/Os 86, 104
    - balance table space allocations 25
    - cache statistics 24
    - collecting I/O events 103
    - detail I/O events 87, 103
    - detail trace summary 80
    - Primary Option Menu selection 103, 104
    - printing I/O reports 106
    - running a system I/O trace 103
    - summarized trace events 86, 87, 104
    - synchronous I/Os 86, 104
    - system-wide traces 102
    - total measurements 104
    - volume summary 23
      - with data sharing 46
  - improving performance
    - tutorial example 67, 112
  - index administration
    - tutorial example
      - access more details 134
      - information for one index 134
- INPUT mode
  - monitor history display
    - tutorial example 34
- interval
  - counts
    - tutorial example 9
  - default timer request 28
- INTERVAL keyword
  - Trace Summary by Time display
    - tutorial example 95
- isolating performance problems 5, 97

## J

- Journal log
  - Enhanced 39
  - finding exception messages 99
  - viewing a subset of messages
    - from current target DB2 39
  - warning messages sent to 28

## K

- key indicators
  - STRAC display
    - tutorial example 72, 97
  - TSTAT display
    - tutorial example 85, 98

## L

- LEVEL keyword
  - tutorial example 83
- LINECNT keyword 141
- LKBWZ view
  - tutorial example 21
- LKBWZSSI view
  - tutorial example 53
- LKCONZ view
  - tutorial example 21
- LKEVENT view
  - tutorial example 19
- LKEVSSI view
  - tutorial example 52
- LKRESD view
  - tutorial example 20, 53
- LKRESNRZ view
  - tutorial example 20, 52
- LKRESZ view
  - tutorial example 20, 52
- Local Buffer Pool Statistics Detail view 17
- location
  - connecting to remote 122, 130
- lock
  - analysis
    - avoiding expensive traces 81
    - detail trace events 81
    - detail trace summary 80
    - global contention 50
    - hot spots in tables 52
    - resource information 20, 52
    - see plans 53
    - timeouts and deadlocks 19, 50
  - contention
    - global 50
    - tutorial example 19, 32
  - pop-up
    - summary 84, 119
    - timeout 84
- Lock Wizard Global Members view 51
- Lock Wizard Global/Local Contention view 51
- Lockout Blocker/Waiter Summary view 21
- Lockout Connection Summary view 21
- Lockout Events view 19
- Lockout Global Blocker Waiter Summary view 53
- Lockout Resource Conflict Detail view 20, 53
- Lockout Resource Number Summary view 20, 52
- Lockout Resource Summary view 20, 52
- log
  - application transfer
    - tutorial example 38, 99
  - data sets
    - displaying current 100
    - displaying history traces 69, 101
    - logging a trace 65
    - printing a trace 88, 106, 136
  - display
    - selecting current target DB2 39
    - tutorial example 38

- log (*continued*)
  - display (*continued*)
    - viewing subsets 39
  - keyword
    - tutorial example 28
- logging
  - DB2 messages
    - selecting current target DB2 39
    - tutorial example 38
    - viewing a subset 39
- LOGTRAC keyword
  - tutorial example 62, 103
- long-running threads
  - tracing 69
- LTRAC
  - display
    - scrolling 71
    - tutorial example 71

## M

- MAINVIEW
  - accessing
    - tutorial example 6
  - Selection Menu
    - tutorial example 6
  - maximum PLOT graph values
    - tutorial example 33, 36
  - maximum warning threshold value
    - starting a monitor 28
- MAXPAGES keyword 141
- messages
  - DB2 38
  - exception
    - displaying 31
    - tutorial example 13
  - MAINVIEW for DB2 38
  - origin identifier 39
  - viewing a subset
    - from current target DB2 39
- MESSAGES option
  - tutorial example 31
- minimum warning threshold value
  - starting a monitor 28
- MODIFY (M) line command
  - tutorial example 29
- modifying requests
  - tutorial example 29
- monitor
  - a data sharing group 41
  - area summary (DMAREAZ) 36
  - definition 26
  - tutorial example
    - defaults 28
    - graph of warning status 26, 32
    - history display 33, 36
    - isolating problems 26
    - list of active timer requests 26, 32
    - looking at different workloads 27

- monitor (*continued*)
  - tutorial example (*continued*)
    - modifying 29
    - multiple requests 26, 27
    - Primary Option Menu selection 7, 26
    - replicating 29
    - resource 27
    - starting 26, 27, 28
    - stopping 29
    - threshold marker 26, 32
    - windows mode views 35
    - workload 27
  - warning summary (DWARN) 35
  - workload objectives (DOBJ) 37
- Monitor Summary by Area view 36
- Monitors in Warning view 35
- multi-index pop-up
  - tutorial example 84
- multiple
  - DB2s
    - tutorial example 8
  - requests
    - automatic start 26, 32
- MVS console
  - warning messages sent to 28

## N

- navigating between traces 70
  - between threads (STRAC display) 82
- NEWPAGE keyword 141

## O

- objective
  - workload
    - monitoring 37
- Objectives Review view 37
- overview
  - of all DB2s
    - tutorial example 8

## P

- package
  - overview (STRAC)
    - tutorial example 75
- page set
  - considerations
    - with data sharing 46
  - views
    - status 22
- Page Set Cache view 24
- Page Set GBP-DEP Status view 49
- Page Set Status view 22
- parameters
  - service request
    - tutorial example 28
  - trace request
    - tutorial example 62, 103

- performance
  - of a bad SQL statement
    - improving 67, 112
  - of an existing application
    - analyzing 67, 112
  - problems
    - isolating 5, 97
- Performance Reporter
  - reports 148
- PF10/22 (display next entry)
  - tutorial example 82, 119
- PF11/23 (display previous entry)
  - tutorial example 82, 119
- PF5/17 (Log Display REPEAT FIND)
  - tutorial example 38
- PF5/17 (Log Display transfer)
  - tutorial example 38, 99
- PF6/18 (GO)
  - tutorial example 34
- PF7/19 (scroll)
  - tutorial example
    - Data Collection Monitor list application 27
    - DB2 Trace Entries display (LTRAC) 71
    - Detail Trace Entry display (DTRAC) 83
    - Trace Summary by Time display (TSUMT) 94
- PF8/20 (scroll)
  - tutorial example
    - Data Collection Monitor list application 27
    - DB2 Trace Entries display (LTRAC) 71
    - Trace Summary by Time display (TSUMT) 94
- plan
  - I/O analysis
    - across data sharing members 53
- PLAN administration
  - tutorial example
    - access more details 131
    - information for one plan 130
- PLAN keyword
  - tutorial example 96, 98
- PLOT
  - use
    - automatic refresh mode 34
    - exceptions only 34
    - tutorial example 33, 36
- PLOTMAX keyword
  - tutorial example 28
- plus (+)
  - qualifier
    - tutorial example 98
- point and shoot
  - tutorial examples
    - BIND-TEXT pop-up 118
    - from DBIOD display 128
    - from DBIOR display 128, 129
    - from DTRAC display 118, 119
    - from DUSER display 122, 123
    - from LTRAC display 71, 85, 86
    - from SQL statement pop-up 78, 126
    - from TSTAT display 86
    - from USERS display 122

- point and shoot (*continued*)
  - tutorial examples (*continued*)
    - to a subset of threads (LTRAC) 96
    - to buffer pool activity 74
    - to current trace log data set 100
    - to database summary 79
    - to DB2 catalog information 128
    - to DBIO display 86, 104, 105
    - to DBIOD display 128
    - to DBIOP display 105
    - to DBIOS display 87
    - to detail entry display (DTRAC) 83, 127
    - to DTRAC display 118
    - to DUSER display 122
    - to elapsed time analysis 73
    - to EXPLAIN PLAN\_TABLE display 122, 123
    - to EXPLAIN pop-up 119
    - to lock and I/O events 80
    - to LOCK-SUMMARY pop-up 119
    - to package details 75
    - to sort summary 81
    - to SQL statement pop-up 78, 84, 126
    - to SQL statement summary 76, 82, 125
    - to SQL statement text 78, 126
    - to STRAC display 71, 97
    - to TSTAT display 85, 98
    - to TSUMP display 96
- pop-up
  - asterisk (\*) indicator 84
  - BIND TEXT
    - tutorial example 84, 118
  - definition vii
  - EXPLAIN
    - tutorial example 84, 119
  - LOCK-SUMMARY
    - tutorial example 84, 119
  - LOCK-TMO
    - tutorial example 84
  - MULTI-INDEX
    - tutorial example 84
  - SQL statement
    - tutorial example 78, 84, 119, 126
  - usage 84
- Primary Option Menu
  - tutorial example
    - access to analyzers, monitors, and traces 7
    - RxD2 112
- printing
  - reports 135
    - offline 143
    - online 136
  - screens
    - series of displays 30
    - single display 30
  - traces
    - SQL prototype 120
    - tutorial example 88, 89, 106, 136, 137

- problems
  - analyze lock and I/O
    - tutorial example 80
  - review critical
    - tutorial example 30
- product
  - line transfer
    - to DB2 catalog information (RxD2) 128
    - to other MAINVIEW products 34
    - to RxD2 112, 121
  - stacking 34
- PROFILE command 39
- prototyping
  - SQL statements 112
  - print results 120
- PSBPGBPZ view
  - tutorial example 48
- PSBPSZ view
  - tutorial example 25
- PSCACHE view
  - tutorial example 24
- PSGBP view
  - tutorial example 49
- PSSTAT view
  - tutorial example 22
- PSVOLPS view
  - tutorial example 47
- PSVOLSSI view
  - tutorial example 46
- PSVOLSZ view
  - tutorial example 23, 46

## Q

- QMF
  - queries 151
- qualifier
  - wild cards
    - asterisk (\*) 98
    - plus (+) 98
- quick start
  - reference card 1

## R

- RANGES keyword
  - tutorial example 28
- recent
  - thread history 94, 100
- reference card
  - MAINVIEW for DB2 1
- refresh
  - monitor history display
    - tutorial example 34
- remote
  - catalogs
    - accessing 130
  - locations
    - connecting to 122, 130

- REPLICATE (R) line command
  - tutorial example 29
- report keywords
  - tutorial example
    - data selection 89, 108, 137
    - formatting options 91, 141
    - SQL prototype 120
- report samples
  - printing 135
  - tutorial examples 91, 108, 109, 141
- report selection
  - batch trace print 89, 108, 137
  - SQL prototype 120
- RESET USERID command
  - authorization
    - for tutorial exercises 59
- resource
  - monitors 27
- response time
  - objectives
    - monitoring 37
- reviewing critical problems
  - tutorial example 30
- reviewing history 15
- RFINDD command 38
- ROLLBACK
  - execute SQL 114, 126
- RST keyword
  - tutorial example 28, 62
- RUNNING mode
  - monitor history display
    - tutorial example 34
- RX command 112, 121
- RX IX (indexname|OBID)
  - command 134
- RX PL (planname)
  - command 130
- RX PT (planname) (owner)
  - command 132
- RX TB (tablename|OBID)
  - command 133
- RxD2
  - accessing from other products 121
  - using with MAINVIEW for DB2 111
    - expand from DBIOD 128
    - expand from DBIOR 129

## S

- S line command
  - tutorial example
    - select a workload monitor 27
- scans
  - by table space and index space 79
- screen
  - printing
    - series of displays 30
    - single display 30

- scrolling
  - tutorial example
    - Data Collection Monitor list application 27
    - DB2 Trace Entries display (LTRAC) 71
    - Detail Trace Entry display (DTRAC) 83
    - Trace Summary by Time display (TSUMT) 94
- selection criteria
  - for workloads
    - starting a monitor 28
- SEQ parameter
  - navigating between trace entries
    - tutorial example 82
- service
  - objective
    - monitoring 37
- service-level objective
  - workload
    - monitoring 37
- session
  - counts
    - tutorial example 9
- severity codes
  - tutorial example 31
- single system image (SSI)
  - contexts
    - defining 42
    - selecting 42
  - using
    - tutorial example 8
- SM application transfer
  - tutorial example 26
- SORT
  - tutorial example
    - by average I/O wait time 104
    - by maximum I/O wait time 104
    - by number of I/Os 105
    - by pages scanned 77
    - by percent I/O wait time 86, 105
    - by user ID 71
    - detail trace summary (STRAC) 81
    - evaluate impact 81
    - lockout events (LKEVENT) 19
    - trace log data sets 69, 101
    - Trace Summary by Time display 95
- SQL
  - action
    - COMMIT 114, 126
    - ROLLBACK 114, 126
  - execute
    - tutorial example 114, 126
  - tutorial example
    - full text of current statement 12, 122
    - printing 120
    - prototyping 112
    - statement pop-up (DTRAC) 84, 119
    - statement pop-up (STRAC) 78, 126
    - statement summary (STRAC) 76, 82, 125
- SQL Counts Detail view 16
- SQLID
  - current 115, 122, 126

- SSI Context Activity Manager view 42
- SSI Context Selection List view 42
- SSI Easy Menu 8, 35
- ST application transfer
  - tutorial example 60, 113
- START
  - keyword
    - default timer request 28
    - default trace request 62
    - tutorial example 103
- Start DB2 Trace Request
  - data entry panel 61
- Start DB2 Workload Monitor Request
  - data entry panel 27
- start time
  - default timer request 28
  - default trace request 62
  - tutorial example 103
- starting
  - monitors
    - tutorial example 26, 27
  - trace
    - tutorial example 60, 103, 113
- status
  - of all DB2s 8
  - of current traces 60, 66
  - of one DB2 15
    - detail view 15
  - of thread completion 72
- Status Detail - Interval view 15
- STDB2 view
  - tutorial example 8, 15
- STDB2D view
  - tutorial example 15
- STEXC view
  - tutorial example 13
- STOP
  - (Z) line command 29
  - keyword
    - default timer request 28
    - default trace request 62
    - tutorial example 103
- stop time
  - default timer request 28
  - default trace request 62
  - tutorial example 103
- stopping
  - monitors 29
- STORAGE keyword
  - default trace request 62
  - I/O trace request 103
- STRAC
  - display
    - tutorial example 72, 97, 138
- STRATE view
  - tutorial example 9
- STSQLD view
  - tutorial example 16

- summary trace
  - authorization ID
    - tutorial example 99
  - connection name
    - tutorial example 99
  - day
    - tutorial example 99
  - location name
    - tutorial example 99
  - plan name
    - tutorial example 96
  - time
    - tutorial example 94, 100
- Summary Trace Entry display
  - tutorial example
    - base section 72, 97
    - buffer pool activity 74
    - database summary 79
    - detail event summaries 76
    - elapsed time 73
    - lock and I/O events 80
    - navigate 82
    - package/DBRM overview 75
    - sort summary 81
    - SQL statement pop-up 78, 126
    - SQL statement summary 76, 82, 125
- SYNC parameter
  - tutorial example 86
- syntax
  - notation vii

## T

- tab points for expanding
  - tutorial examples
    - BIND-TEXT pop-up 118
    - from DBIOD display 128
    - from DBIOR display 128, 129
    - from DTRAC display 118, 119
    - from DUSER display 122, 123
    - from LTRAC display 71, 85, 86
    - from SQL statement pop-up 78, 126
    - from TSTAT display 86
    - from USERS display 122
    - to a subset of threads (LTRAC) 96
    - to buffer pool activity 74
    - to current trace log data set 100
    - to database summary 79
    - to DB2 catalog information 128
    - to DBIO display 86, 104
    - to DBIOD display 128
    - to DBIOP display 105
    - to DBIOS display 87
    - to detail entry display (DTRAC) 83, 127
    - to DTRAC display 118
    - to DUSER display 122
    - to elapsed time analysis 73
    - to EXPLAIN PLAN\_TABLE display 122, 123
    - to EXPLAIN pop-up 119

- tab points for expanding (*continued*)
  - tutorial examples (*continued*)
    - to lock and I/O events 80
    - to LOCK-SUMMARY pop-up 119
    - to package details 75
    - to sort summary 81
    - to SQL statement pop-up 78, 84, 126
    - to SQL statement summary 76, 82, 125
    - to SQL statement text 78, 126
    - to STRAC display 71, 97
    - to TSTAT display 85, 98
    - to TSUMP display 96
- table administration
  - tutorial example
    - access more details 133
    - information for one table 133
- table space
  - allocation
    - balancing 25
  - catalog display 128
  - qualifier panel 129
- target
  - when hyperlinking to RxD2 122, 130
- testing
  - an application 67, 112
  - an SQL statement 112, 114
  - compare results 118
  - tips 67, 112
- THDACTV view
  - tutorial example 11
- thread activity
  - current 11
    - multiple DB2s 13
- thread history
  - graphic display 94, 100
- threshold
  - marker 26, 32
    - monitor history display 33
  - warning message 28, 32
- time
  - command
    - tutorial example 15
  - of exception
    - locate 38
- TIME keyword
  - tutorial example 96, 98
- title
  - defining 62, 103
- TITLE keyword
  - tutorial example 62, 103
- trace
  - tutorial example
    - accessing EXPLAIN 124
    - accessing services 70
    - activating 66
    - additional options 62
    - analyzing DB2 workload 93
    - avoiding expensive lock event traces 81
    - BIND-TEXT pop-up 118
- trace (*continued*)
  - tutorial example (*continued*)
    - buffer pool activity 74
    - collecting I/O events 103
    - database summary 79
    - defaults 61
    - detail entry display (DTRAC) 83, 118, 127
    - detail event summaries 76
    - detail events 64, 103
    - elapsed time 73
    - error messages 66
    - exception filters 63
    - EXPLAIN pop-up 119
    - history 136
    - I/O analysis 86, 102
    - improving performance 67, 112
    - list of current 60, 69
    - list of historical data sets 69, 88, 101, 136
    - list of trace entries (LTRAC) 71
    - lock and I/O events 80
    - LOCK-SUMMARY pop-up 119
    - logging 65
    - long-running threads 69
    - managing log data sets 88, 136
    - narrowing 63
    - navigating 70, 82
    - package/DBRM overview 75
    - Primary Option Menu selection 7, 60, 69
    - printing 88, 106, 136
    - qualifying 62
    - sort summary 81
    - SQL statement pop-up 78, 84, 126
    - SQL statement summary 76, 82, 125
    - starting 60, 103, 113
    - summary by authorization ID 99
    - summary by connection name 99
    - summary by DDF location 99
    - summary by plan 96
    - summary by relative day 99
    - summary by time 94, 100
    - summary entry display (STRAC) 72, 97
    - summary of all threads (TSTAT) 85, 98
    - testing applications 67, 112
    - totals (TSTAT) 85, 98
    - tuning an application 59, 69
  - Trace Statistics display
    - tutorial example
      - base section 85, 98
  - trace summary
    - by authorization ID
      - tutorial example 99
    - by connection name
      - tutorial example 99
    - by location
      - tutorial example 99
    - by plan name
      - tutorial example 96
    - by relative day
      - tutorial example 99
    - by time
      - tutorial example 94, 100

- TRANSFER command
  - windows to full-screen mode
    - tutorial example 16
- transferring applications
  - tutorial examples 28, 30, 101
    - CT command 113
    - LOG command 38, 99
    - SM command 26
    - ST command 60, 113
  - to full-screen mode 16
- transferring between products
  - to other MAINVIEW products 34
  - to RxD2 112, 121
    - expand from DBIOD 128
    - expand from DBIOR 129
- TRBUFF
  - Detail Trace Options panel
    - tutorial example 64
- trend
  - of monitor history values 33, 36
- TRSIZE
  - Detail Trace Options panel
    - tutorial example 64
- TSO id operand
  - send warning messages to a user 28
- TSTAT
  - display
    - tutorial example 85, 98
- tuning
  - an application with trace 59, 69
  - group buffer pools 54
  - information
    - tutorial example 48
  - SQL prototyping 112
  - print results 120
- TYPE
  - keyword
    - tutorial example 62, 103

**U**

- user
  - Detail Status display
    - tutorial example 11, 122
  - warning messages sent to 28
- UTRAC display
  - view detail events as they occur
    - tutorial example 69

**V**

- volume
  - I/O analysis
    - tutorial example 23
    - with data sharing 46
- Volume I/O SSI Summary - Session view 46
- Volume I/O Summary - Session view 23, 46
- Volume Page Sets view 47

**W**

- W
  - marker
    - graph of warning status 26, 32
    - monitor history display 33
- warning
  - all monitors 35
  - automatic 26, 28, 32
  - message
    - current list 28
    - hyperlink 9
    - to a user 28
    - to Journal log 28
    - to the MVS console 28
    - triggering AutoOPERATOR EXECs 28
    - tutorial example 13
  - overview 32
  - status 26, 32
  - threshold marker
    - graph of warning status 26, 32
    - monitor history display 33
  - threshold value
    - starting a monitor 28
- WHERE
  - command
    - tutorial example 23
- WIDTH keyword 141
- WIF keyword
  - tutorial example 28
- WIN keyword
  - tutorial example 28
- window
  - information line
    - tutorial example 8
- wizards
  - tutorial example
    - data sharing 47
    - global lock 50
- WLIM keyword
  - tutorial example 28
- WMSG keyword
  - tutorial example
    - starting a monitor 28
- workload
  - objectives
    - monitoring 37
    - service-level objective
      - monitoring 37
- workload history
  - analyze
    - navigate through trace displays 96
    - past 100
    - recent 94
- workload monitors 27
- workload selection criteria
  - starting a monitor 28
- WRAP keyword
  - default trace request 62

WTO (write-to-operator operand)  
  send messages to MVS console 28

WVAL keyword  
  tutorial example  
  starting a monitor 28

WZDSBOPT view  
  tutorial example 55

WZDSDEP view  
  tutorial example 47

WZDSGBR view  
  tutorial example 56

WZDSGBRD view  
  tutorial example 56

WZDSGBSA view  
  tutorial example 57

WZDSGMEM view  
  tutorial example 54

WZLKGCON view  
  tutorial example 51

WZLKGMEM view  
  tutorial example 51

WZLKGOPT view  
  tutorial example 50

## **X**

X ON|OFF command  
  tutorial example 26, 34

## **Z**

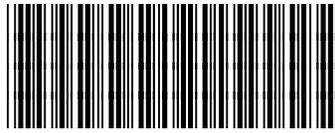
Z line command (STOP)  
  tutorial example 29



---

## Notes

---



\*100025884\*