MAINVIEW® for DB2®

User Guide

Volume 3: Traces

Version 7.1

June 12, 2001

bmcsoftware

Copyright © 1989-2001 BMC Software, Inc., as an unpublished work.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 UNPUBLISHED–RIGHTS RESERVED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES.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

Part 1. Using a Trace 1

Chapter 1. Requesting a Trace 3
Starting a Trace 3
Displaying a Trace 4
Printing a Trace 4
Stopping a Trace 5

Chapter 2. Collecting Trace Data 7
Trace Authorization 8
Detail Trace Data Collection 8
Detail Trace Online Data Storage 9
Trace Logging 11
ATRAC—DB2 Application Trace 13
Starting a Trace 13
Start DB2 Trace Request Panel 13
Additional Trace Options Processing 18
Selecting Trace Data for Retention 18
Exception Filter Keyword Descriptions 19
Starting a Detail Trace 21
Detail Trace Options 21
Recommendations When Starting a Detail Trace 25
Specifying Trace Logging Options 28
Log Switching Considerations 31
Replicating a Trace Request 31

Chapter 3. Controlling Current Traces (Option 4) 33
Primary Commands 34
Line Commands 35
Displaying Trace Entries 35
Showing Trace Options 35
Modifying a Trace 37
Switching Log Data Sets 38
Stopping a Trace 38
Purging a Trace 38
Replicating a Trace 38
Quiescing Logging 38

Chapter 4. Managing History Traces 39
The Trace Directory 39
Displaying Historical Trace Data 40
Target Specification 40
Using the History Traces Application (Option 5) 41
Field Definitions 42
Primary Commands 43
Line Commands 45
Batch Maintenance Jobs 48
Trace Log Data Set Jobs 48
Defining a Trace Log Data Set 48
Archiving a Trace Log Data Set 48
Restoring an Archived Trace Log Data Set 48
Printing a Trace Log Data Set 49

Chapter 5. Printing a Trace 51
Overview 51
Types of Reports 51
Summary Traces 51
Detail Traces 52
Detail Traces with I/O Events 52
Primary Input - Trace Log Data Set(s) (TLDS) 53
Printing from Archived TLDSs 53
Secondary Input - SMF or GTF 53
Output Formats 54
Controls 54
JCL 55
Output Format 58
Controls for Reporting 59
Structure of a Report Request 59
Request Keywords 60
Identification Options 61
Selection Options 62
Report Options 65
Special Report Summarization Options for DBIO 69
Formatting Options 70
Default Entry 71
Return Codes 71
Examples of Output 72
Accounting Summary Report 73
Workload Summary by Time 74
Accounting Detail—Long 75
Accounting Detail—Short 77
SQL Statement Summary 78
Thread Detail Events 79
SQL Text and EXPLAIN 80
Lock and EDM Pool Failures 81
I/O Summary by DB/TS 82

Chapter 6. Solving Problems 83

Part 2. Application Trace 89

Chapter 7. Introduction 93

Chapter 8. Application Trace Considerations 95
Data Collection 95
Display Level Options 97
Usage Examples 98
(Application Developer) Test Application Performance 98
(DBA) Review SQL Before Production Use 100
(DBA) Review Application Performance Periodically 101
(DBA) Investigate Problem Plans 102
(DBA) Analyze Index Usage and Database Access Paths 102
(DBA) Trace QMF and Ad Hoc SQL Usage 103
(DBA) Analysis of Short-Term Workload History 103
(DBA) Analyze I/O by Page Set 105
(DBA) Analyze Lock Contention 106
(System Programmer) Check New Production Application Performance 107
(System Programmer) Set Up Ongoing Exception Traces per Workload Type 108

Chapter 9. Application Trace Displays 109
Trace Display Summary 109
Navigating Between Trace Displays 111

Chapter 10. Viewing a List of Completed Threads 115
LTRAC—DB2 Trace Entries 115

Chapter 11. Viewing Summary Data for All Trace Entries 121
TSTAT—Trace Statistics 121
Summary Statistics (Base Section) 130
Accounting Sections 139
Elapsed Time Analysis 139
SQL Statement Execution Counts 144
Buffer Pool Activity 148
Lock Activity 151
Parallelism 154
Routines 157
DDF Summary 158
Detail Trace Summary Sections 161
Database Summary 161
Database Lock and I/O Summary 162
Sort Summary 164

Chapter 12. Viewing Workload History Summaries 165
TSUMx—Trace Summary Displays 165
Display Fields 165
Service Messages 167
Scrolling 167
Using EXPAND 167
Displaying Graphics 168
Sorting the Display 169
Selecting a Subset of Trace Entries 171
TSUMA—Trace Summary by Authorization ID (AUTHID) 172
TSUMC—Trace Summary by Connection Name (CONNECT) 174

TSUML—Trace Summary by Location Name (LOCATION) 176
TSUMP—Trace Summary by Plan Name (PLAN) 178
TSUMR—Trace Summary by Correlation ID (CORR-ID) 180
TSUMT—Trace Summary by Time (INTERVAL START) 182

Chapter 13. Viewing Summary Data for a Specific Thread 185
STRAC—Summary Trace Entry 185
Thread Identifiers and Status Summary (Base Section) 198
Accounting Sections 210
Environmental Indicators 210
Elapsed Time Analysis 211
SQL Statement Execution Counts 216
Buffer Pool Activity 220
Lock Activity 223
Parallelism 226
Routines 229
DDF Summary 233
Package/DBRM Overview 237
Package/DBRM Pop-up 239
Detail Trace Summary Sections 242
SQL Summary 242
SQL Statement Pop-up 246
Database Summary 249
Database Lock and I/O Summary 251
Sort Summary (Detail Trace Only) 253

Chapter 14. Viewing All Events for a Specific Transaction 255
DTRAC—Detail Trace Entry 255
Display Fields 258
Group SQL 258
Event Data 259
Event Pop-Up Displays 274
Allocate Cursor Pop-Up (IFCID 273) 275
Associate Locators Pop-Up (IFCID 272) 276
Authorization Failure Pop-Up (IFCID 140) 277
Authorization Translation Pop-Up (IFCID 169) 278
Bind Pop-Up (IFCID 108/109) 279
Bind Text Pop-Up (IFCID 063) 283
Claim Pop-Up (IFCID 211) 284
Claim Wait Pop-Up (IFCID 215/216) 286
COMMIT-LSN Pop-Up (IFCID 218) 287
Conversation Queued Pop-Up (IFCID 167) 288
Conversation Request Pop-Up (IFCID 159) 289
Deadlock Pop-Up (IFCID 172) 290
Distributed SQL Pop-Up (IFCID 168) 293
Drain Pop-Up (IFCID 212) 294
Drain Wait Pop-Up (IFCID 213/214) 296
EXPLAIN Pop-Up (IFCID 022) 298
Free Pop-Up (IFCID 110/111) 314
Hybrid Join Failure Pop-Up (IFCID 190) 315
Lock Request Pop-Up (IFCID 021) 316
Lock Summary Pop-Up (IFCID 020) 318
Lock Suspension Pop-Up (IFCID 044/045) 320
Lock Timeout/Deadlock Pop-Up (IFCID 054) 322

Page P-Lock Pop-Up (IFCID 259) 323
Package Allocation Pop-Up (IFCID 177) 325
Parallel Degree Pop-Up (IFCID 221) 327
Parallel Elapsed Pop-Up (IFCID 222) 330
Parallel Group Pop-Up (IFCID 231) 332
Plan Allocation Pop-Up (IFCID 112) 334
P-Lock Pop-Up (IFCID 251) 335
Read I/O Pop-Up (IFCID 006/007) 339
Requesting Agent Data Pop-Up (IFCID 183) 340
Requesting Site—DRDS Pop-Up (IFCID 157) 342
Requesting Site—DTM (IFCID 162) 343
Requesting Site—VTAM Pop-Up (IFCID 160) 344
RID List Pop-Up (IFCID 125) 346
Scan Pop-Up (IFCID 017/018) 348
Serving Site—DRDS Pop-Up (IFCID 158) 350
Serving Site—DTM Pop-Up (IFCID 163) 351
Serving Site—VTAM Pop-Up (IFCID 161) 352
Sort Pop-Up (IFCID 95/96) 354
SQL Statement Pop-Up 356
Stored Procedures Entry Pop-Up (IFCID 233) 359
Stored Procedures Exit Pop-Up (IFCID 233) 360
Synchronous Write Pop-Up (IFCID 008/009) 361
Table Check Constraint Pop-Up (IFCID 305) 362
Temporary Table Pop-Up (IFCID 311) 364
Timeout Pop-Up (IFCID 196) 366
Trigger Entry Pop-Up (IFCID 325) 369
Trigger Exit Pop-Up (IFCID 325) 371
User-Defined Function Pop-Up (IFCID 324) 373
Utility Pop-Up 375
VTAM Exit Pop-Up (IFCID 164) 377
VTAM Macro Pop-Up (IFCID 165) 378
Wait I/O Pop-Up (IFCID 127/128) 379
UTRAC—User Detail Trace 380
Display Fields 381

Chapter 15. Viewing I/O Activity 383
DBIOx—I/O Analysis (Trace) 383
Display Fields 384
Description 385
Service Message(s) 385
Using EXPAND 385
Sorting the Display 386
Selecting a Subset of I/O Events 386
DBIO—I/O Analysis by Database / Table Space (DB/TS) 388
DBIOA—I/O Analysis by Authorization ID (AUTHID) 390
DBIOF—I/O Analysis by Buffer Pool (BPOOL) 391
DBIOC—I/O Analysis by Connection Name (CONNECT) 392
DBIOL—I/O Analysis by Location (LOCATION) 393
DBIOP—I/O Analysis by Plan (PLAN) 394
DBIOK—I/O Analysis by Package / Program (PKG/PGM) 396
DBIOS—I/O Analysis by SQL Statement (SQL STMT) 398
DBIOT—I/O Analysis by Time (INTERVAL START) 400

Part 3. Appendixes/Glossary 403

Appendix A. Lock Type Table 405

Appendix B. IFCID Table (ATRAC Data Collection) 409
TYPE=D Data Collection 410
TYPE=SQL Data Collection 412
TYPE=SCAN Data Collection 413
TYPE=IO Data Collection. 413
TYPE=LOCK Data Collection 414
TYPE=DDF Data Collection. 414
TYPE=DDFVTAM Data Collection 415

Glossary 417

Part 4. Index 427

Index 429

Figures

1. Organization of MAINVIEW Documentation xxi
2. Detail Trace Buffer Storage 9
3. Start DB2 Trace Request Panel 14
4. DB2 Trace Exception Filters Panel 18
5. Detail Trace Options Panel 21
6. Trace Logging Specification Panel 28
7. Current Traces Application 33
8. Show Trace Logging Specification Panel 35
9. History Traces Application (Before Scrolling Right) 41
10. History Traces Application (After Scrolling Right) 41
11. Panel for New Primary Command 44
12. Batch Trace Print Panel 45
13. Batch Trace Print Job Statement 46
14. Trace Directory Purge Confirmation Panel 47
15. JCL to Print a Trace (DZTBTRAC) 55
16. Report Header 58
17. Accounting Summary Report 73
18. Workload Summary by Time 74
19. Accounting Detail—Long—Part I 75
20. Accounting Detail—Long—Part II 76
21. Accounting Detail—Short 77
22. SQL Statement Summary 78
23. Thread Detail Events 79
24. SQL Text and EXPLAIN 80
25. Lock and EDM Pool Failures 81
26. I/O Summary by DB/TS 82
27. Trace Display Service Access 113
28. DB2 Trace Entries Panel 115
29. Summary Statistics for all Trace Buffer Entries (Base Section) 121
30. TSTAT Elapsed Time (ELAPSED) Section 122
31. TSTAT SQL Statement Execution Counts (SQLCOUNTS) Section 122
32. TSTAT Buffer Pool (BPOOL) Section 123
33. TSTAT Locks (LOCKS) Section 124
34. TSTAT Parallelism (PRLL) Section 124
35. TSTAT Routines (RTN) Section 125
36. TSTAT DDF Summary (DDF) Section 125
37. TSTAT Database Summary Section 125
38. TSTAT Database Lock and I/O Summary Section 126
39. TSTAT Sort Summary Section 126
40. TSTAT Elapsed Time Analysis (ELAPSED) Section 139
41. TSTAT SQL Statement Execution Counts (SQLCOUNTS) Section 144
42. TSTAT Buffer Pool (BPOOL) Section 148
43. TSTAT Locks (LOCKS) Section 151
44. TSTAT Parallelism (PRLL) Section 154
45. TSTAT Routines (RTN) Section 157
46. TSTAT DDF Summary (DDF) Section 158
47. TSTAT Database Summary Section 161
48. TSTAT Database Lock and I/O Summary Section 162
49. TSTAT Sort Summary Section 164
50. Trace Summary Display Fields 165
51. Sample Trace Summary Display with GRAPH=AVG 168

52. Sample Trace Summary Display with GRAPH=TOT 169
53. Trace Summary by Authorization ID 172
54. Trace Summary by Connection Name 174
55. Trace Summary by Location Name 176
56. Trace Summary by Plan Name 178
57. Trace Summary by Correlation ID 180
58. Trace Summary by Time 182
59. Summary Trace Entry Panel (Base Section) 185
60. STRAC Environmental (ENV) Section 185
61. STRAC Elapsed Time Analysis (ELAPSED) Section 186
62. STRAC SQL Statement Execution Counts (SQLCOUNTS) Section 186
63. STRAC Buffer Pool Activity (BPOOL) Section 187
64. STRAC Lock Activity (LOCKS) Section 188
65. STRAC Parallelism (PRLL) Section 188
66. STRAC Routines (RTN) Section 189
67. STRAC DDF Summary (DDF) Section 190
68. STRAC Package/DBRM Overview (PKG) Section 190
69. STRAC Package/DBRM Pop-up Display 191
70. STRAC SQL Summary Section 192
71. STRAC SQL Statement Pop-up Display 193
72. STRAC Database Summary Section 193
73. STRAC Database Lock and I/O Summary Section 194
74. STRAC Sort Summary Section 194
75. STRAC Environmental (ENV) Section 210
76. STRAC Elapsed Time Analysis (ELAPSED) Section 211
77. STRAC SQL Statement Execution Counts (SQLCOUNTS) Section 216
78. STRAC Buffer Pool Activity (BPOOL) Section 220
79. STRAC Lock Activity (LOCKS) Section 223
80. STRAC Parallelism (PRLL) Section 226
81. STRAC Routines (RTN) Section 229
82. STRAC DDF Summary (DDF) Section 233
83. STRAC Package/DBRM Overview (PKG) Section 237
84. STRAC Package/DBRM Pop-up Display 239
85. STRAC SQL Summary Section 243
86. STRAC SQL Statement Pop-up Display 246
87. STRAC Database Summary Section 249
88. STRAC Database Lock and I/O Summary Section 251
89. STRAC Sort Summary Section 253
90. Detail Trace Display for a Transaction (Level 1) - User Requests 255
91. Detail Trace Display for a Transaction (Level 2) - DB2 Requests 255
92. Detail Trace Display for a Transaction (Level 3) - DB2 Processing Events 256
93. DTRAC Display Fields 258
94. DTRAC Allocate Cursor Pop-Up Display 275
95. DTRAC Associate Locators Pop-Up Display 276
96. DTRAC Authorization Failure Pop-Up Display 277
97. DTRAC Authorization Translation Pop-Up Display 278
98. DTRAC Bind Pop-Up Display 279
99. DTRAC Bind Text Pop-Up Display 283
100. DTRAC Claim Pop-Up Display 284
101. DTRAC Claim Wait Pop-Up Display 286
102. DTRAC COMMIT-LSN Pop-Up Display 287
103. DTRAC Conversation Queued Pop-Up Display 288
104. DTRAC Conversation Request Pop-Up Display 289
105. DTRAC Deadlock Pop-Up Display 290

106. DTRAC Distributed SQL Pop-Up Display 293
107. DTRAC Drain Pop-Up Display 294
108. DTRAC Drain Wait Pop-Up Display 296
109. DTRAC EXPLAIN Pop-Up Display 298
110. DTRAC Free Pop-Up Display 314
111. DTRAC Hybrid Join Failure Pop-Up Display 315
112. DTRAC Lock Request Pop-Up Display 316
113. DTRAC Lock Summary Pop-Up Display 318
114. DTRAC Lock Suspension Pop-Up Display 320
115. DTRAC Lock Timeout/Deadlock Pop-Up Display 322
116. DTRAC Page P-Lock Pop-Up Display 323
117. DTRAC Package Allocation Pop-Up Display 325
118. DTRAC Parallel Degree Pop-Up Display 327
119. DTRAC Parallel Elapsed Pop-Up Display 330
120. DTRAC Parallel Group Pop-Up Display 332
121. DTRAC Plan Allocation Pop-Up Display 334
122. DTRAC P-Lock Pop-Up Display 335
123. DTRAC Read I/O Pop-Up Display 339
124. DTRAC Requesting Agent Data Pop-Up Display 340
125. DTRAC Requesting Site—DRDS Pop-Up Display 342
126. DTRAC Requesting Site—DTM Pop-Up Display 343
127. DTRAC Requesting Site—VTAM Pop-Up Display 344
128. DTRAC RID List Pop-Up Display 346
129. DTRAC Scan Pop-Up Display 348
130. DTRAC Serving Site—DRDS Pop-Up Display 350
131. DTRAC Serving Site—DTM Pop-Up Display 351
132. DTRAC Serving Site—VTAM Pop-Up Display 352
133. DTRAC Sort Pop-Up Display 354
134. DTRAC SQL Statement Pop-Up Display 356
135. DTRAC Stored Procedures Entry Pop-Up Display 359
136. DTRAC Stored Procedures Exit Pop-Up Display 360
137. DTRAC Synchronous Write Pop-Up Display 361
138. DTRAC Table Check Constraint Pop-Up Display 362
139. DTRAC Temporary Table Pop-Up Display 364
140. DTRAC Timeout Pop-Up Display 366
141. DTRAC Trigger Entry Pop-Up Display 369
142. DTRAC Trigger Exit Pop-Up Display 371
143. DTRAC User-Defined Function Pop-Up Display 373
144. DTRAC Utility Pop-Up Display 375
145. DTRAC VTAM Exit Pop-Up Display 377
146. DTRAC VTAM Macro Pop-Up Display 378
147. DTRAC Wait I/O Pop-Up Display 379
148. User Detail Trace Panel 380
149. I/O Analysis Display Fields 384

Tables

1. Exception Filter Keyword Descriptions. 19
2. Modifiable Logging Options 37
3. Identification Keywords. 61
4. Selection Keywords 62
5. Report Keywords 65
6. Special Summarization Keywords for DBIO 69
7. Formatting Keywords 70
8. Application Trace Services 93
9. Detail Trace Event Data. 259
10. Lock Types. 405
11. TYPE=D (DETAIL) IFCID Data Collection. 410
12. TYPE=SQL IFCID Data Collection 412
13. TYPE=SCAN IFCID Data Collection. 413
14. TYPE=IO IFCID Data Collection 413
15. TYPE=LOCK IFCID Data Collection 414
16. TYPE=DDF IFCID Data Collection 414
17. TYPE=DDFVTAM IFCID Data Collection 415

About This Book

This book describes how to use the online functions of MAINVIEW for DB2. It can be used by the DB2 database administrator, system programmer, or performance analyst to control resource usage and performance. Before using this book, you must be familiar with the IBM DB2 program product.

The batch reporting functions are described in the *MAINVIEW for DB2 Performance Reporter User Guide*.

How This Book Is Organized

This book is organized into three volumes:

- Volume 1: Views
- Volume 2: Analyzers/Monitors
- Volume 3: Traces

These parts are included in Volume 1:

- An overview of how to use MAINVIEW for DB2. It includes
  - An introduction describing the major functions of MAINVIEW for DB2
  - Instructions for logging on to MAINVIEW for DB2
- How to use views and wizards
- How to manage views
- A description of the views you can use to analyze DB2 performance

These parts are included in Volume 2:

- An overview of how to use the MAINVIEW for DB2 services that run in full-screen mode. It includes
  - A description of the Primary Option Menu and how to access the MVDB2 functions
  - The DB2 analyzer, monitor, and trace display services and their parameters organized alphabetically (in a tabular format with page references to the detailed service descriptions)
  - A description of the analyzers, monitors, and traces, and how they are used
  - The MVDB2 general commands and services
- The DB2 resource analyzer display services
- The DB2 resource and workload monitor data collection services

These parts are included in Volume 3:

- How to use the trace facility

- The DB2 application trace display services

Appendixes, a glossary, and an index are included in all volumes.

Throughout the body of this document, MVDB2 refers to MAINVIEW for DB2. The DMR acronym for the product is used occasionally in this book and in many online panels and messages.

Conventions Used in This Book

The following syntax notation is used in this manual. Do not type the special characters.

- Brackets [ ] enclose optional parameters or keywords.
- Braces { } enclose a list of parameters; one must be chosen.
- A vertical line | separates alternative options; one can be chosen.
- An <u>underlined</u> parameter is the default.
- AN ITEM IN CAPITAL LETTERS indicates exact characters; usage can be all uppercase or lowercase.
- Items in lowercase letters are values you supply.

Recommended Reading

The following books are referenced in this edition:

- *DB2 Administration Guide* from IBM
- *DB2 Command and Utility Reference* from IBM
- *DB2 Diagnosis Guide and Reference* from IBM
- *CICS/ESA Problem Determination Guide* from IBM
- *MVS/ESA Application Development Macro Reference* from IBM
- *MAINVIEW® Solutions Guide*
- *Using MAINVIEW®*
- *Quick Start with MAINVIEW®*
- *OS/390 and z/OS Installer Guide*
- *MAINVIEW® Installation Requirements Guide*
- *MAINVIEW® Common Customization Guide*
- *MAINVIEW® Alternate Access Implementation and User Guide*
- *Implementing Security for MAINVIEW® Products*
- *MAINVIEW® Administration Guide*
- *MAINVIEW® for DB2® Release Notes*
- *Getting Started with MAINVIEW® for DB2® and RxD2™*
- *MAINVIEW® for DB2® User Guide Volume 1: Views*
- *MAINVIEW® for DB2® User Guide Volume 2: Analyzers/Monitors*
- *MAINVIEW® for DB2® User Guide Volume 3: Traces*

- *MAINVIEW® for DB2® Performance Reporter User Guide*
- *RxD2™ User Guide*
- *MAINVIEW® for DB2® Customization Guide*

Related Reading

This book is included as part of the MAINVIEW library, which documents all your MAINVIEW products and the tasks associated with using these products.

MAINVIEW Library

The MAINVIEW library documents these products:

- CMF® MONITOR Online (CMF)
- IMSPlex System Manager™ (IPSM)
- MAINVIEW® Alarm Manager (MVALARM)
- MAINVIEW® AutoOPERATOR™ (AO)
- MAINVIEW® Explorer
- MAINVIEW® FOCAL POINT™
- MAINVIEW® for CICS (MVCICS)
- MAINVIEW® for DB2® (MVDB2)
- MAINVIEW® for DBCTL (MVDBC)
- MAINVIEW® for IMS (MVIMS)
- MAINVIEW® for IP (MVIP)
- MAINVIEW® for MQSeries (MVMQS)
- MAINVIEW® for OS/390 (MVMVS)
- MAINVIEW® for UNIX System Services (MVUSS)
- MAINVIEW® for VTAM (MVVTAM)
- MAINVIEW® for Websphere
- MAINVIEW® VistaPoint™ (MVVP)
- Plex Manager (PLEXMGR)

The MAINVIEW library is organized into these three categories:

- Installer documentation
- Administrator documentation
- User documentation

Each book within these categories contains information about specific types of tasks. The following figure shows how this book relates to the other books in the MAINVIEW library.

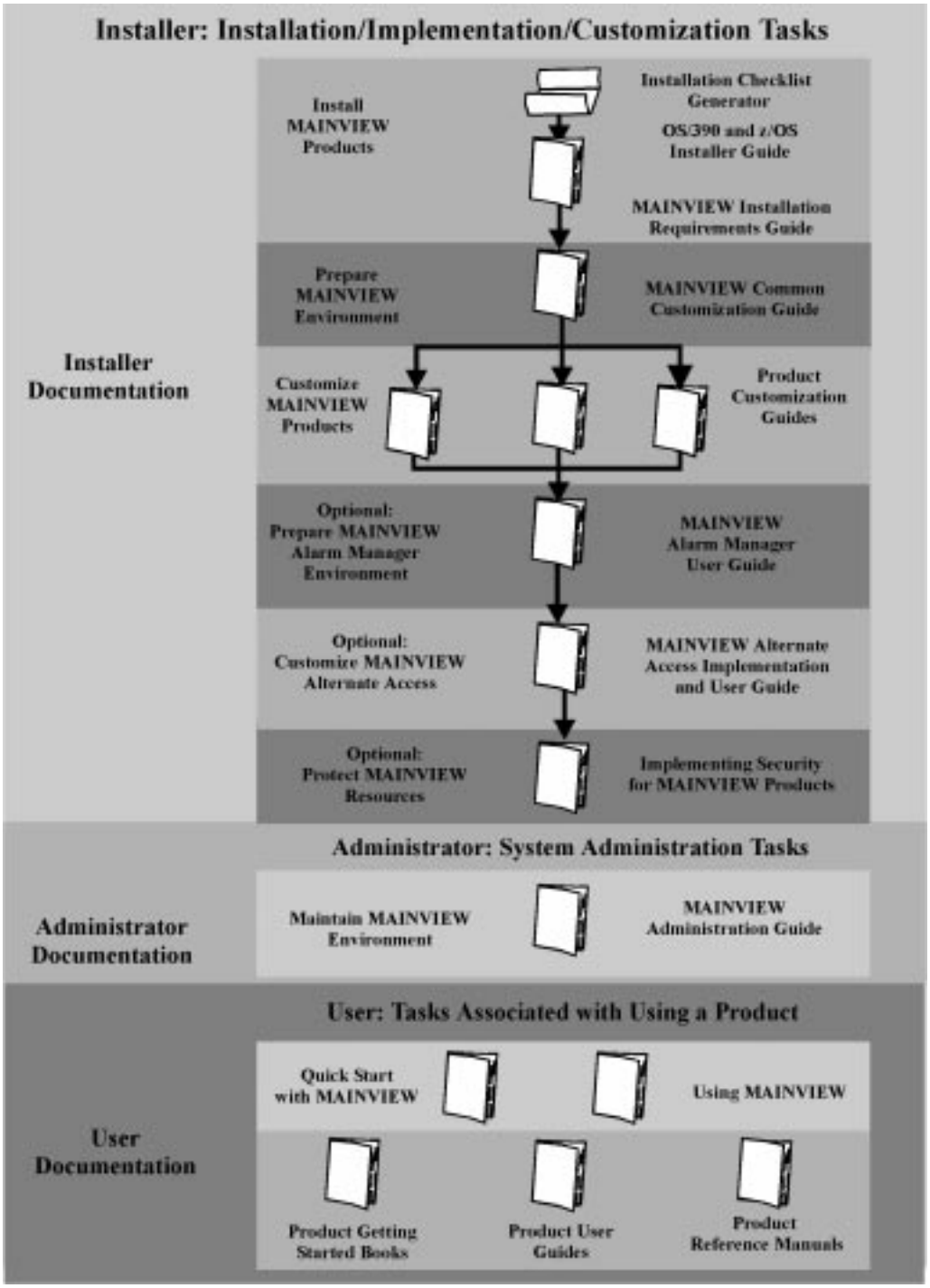

Figure 1. Organization of MAINVIEW Documentation

The following books in the MAINVIEW library are used with all MAINVIEW products:

- *Quick Start with MAINVIEW* gives a brief overview to help you quickly get started using all your MAINVIEW products.
- *Using MAINVIEW* gives a more detailed description of how to use your MAINVIEW products. If you have more than one MAINVIEW product, this book will help you understand how all your MAINVIEW products work. This book will also help you use your products together and take advantage of the integration of all MAINVIEW products.
- The *OS/390 and z/OS Installer Guide* and *MAINVIEW Installation Requirements Guide* give instructions for basic installation of the product libraries.
- The *MAINVIEW Common Customization Guide* and the *MAINVIEW Administration Guide* provide customization and administration instructions for all MAINVIEW products.
- The *MAINVIEW Alternate Access Implementation and User Guide* describes how to use the MAINVIEW Alternate Access component. This component provides EXCP and VTAM communication to BMC Software products through ISPF without requiring a TSO subsystem to be active.
- The *MAINVIEW Alarm Manager User Guide* describes how to generate alarms when thresholds from MAINVIEW product views are exceeded.
- *Implementing Security for MAINVIEW Products* describes how to implement security for MAINVIEW for DB2 with the external security manager installed at your site.
- The *MAINVIEW Explorer Implementation and User Guide* explains how to install and use MAINVIEW Explorer, which provides access to MAINVIEW products from a Web browser running on a Windows workstation.

MAINVIEW for DB2 Library

The following documentation gives specific information about the MAINVIEW for DB2 and RxD2 products:

- *MAINVIEW for DB2 Release Notes* summarize the new features in this release of MAINVIEW for DB2 and RxD2. These notes enable you to quickly see what is new.
- The *MAINVIEW for DB2 Master Index* includes index entries for all the product-specific books in the MAINVIEW for DB2 library. This index shows where you can find information for specific topics in individual MAINVIEW for DB2 and RxD2 manuals.
- *Getting Started with MAINVIEW for DB2 and RxD2* is an introduction for new users of these products. The book helps you use these products to solve problems more effectively in a short time.
- The *MAINVIEW for DB2 User Guide* (Volumes 1, 2, and 3) describes how to use the online views, analyzer, monitor, and trace services for the DB2 database administrator, system programmer, or performance analyst.
- The *MAINVIEW for DB2 Performance Reporter User Guide* describes how to create statistical batch reports about application activity and DB2 performance and resource usage for the database administrator, system programmer, or performance analyst.

- The *RxD2 User Guide* describes how to install and use the RxD2 product, which provides access to DB2 from REXX. This product also provides tools to query the DB2 catalog, issue dynamic SQL, test DB2 applications, analyze EXPLAIN data, generate DDL or DB2 utility JCL, edit DB2 table spaces, perform security administration, and much more. MAINVIEW for DB2 contains numerous hyperlinks to RxD2.
- The *MAINVIEW for DB2 Customization Guide* provides product-specific tailoring instructions. Use this book in conjunction with the *MAINVIEW Installation Requirements Guide* and the *MAINVIEW Common Customization Guide* during the customization process.
- Online tutorials are available by selecting option T from the MAINVIEW for DB2 Primary Option Menu or by pressing HELP (PF1/13) from the product application panels.

Part 1. Using a Trace

This part describes how to use the trace facility.

Chapter 1. Requesting a Trace . . . 3
Starting a Trace . . . 3
Displaying a Trace . . . 4
Printing a Trace . . . 4
Stopping a Trace . . . 5

Chapter 2. Collecting Trace Data . . . 7
Trace Authorization . . . 8
Detail Trace Data Collection . . . 8
Detail Trace Online Data Storage . . . 9
Trace Logging . . . 11
ATRAC—DB2 Application Trace . . . 13
Starting a Trace . . . 13
Start DB2 Trace Request Panel . . . 13
Additional Trace Options Processing . . . 18
Selecting Trace Data for Retention . . . 18
Exception Filter Keyword Descriptions . . . 19
Starting a Detail Trace . . . 21
Detail Trace Options . . . 21
Recommendations When Starting a Detail Trace . . . 25
Specifying Trace Logging Options . . . 28
Log Switching Considerations . . . 31
Replicating a Trace Request . . . 31

Chapter 3. Controlling Current Traces (Option 4) . . . 33
Primary Commands . . . 34
Line Commands . . . 35
Displaying Trace Entries . . . 35
Showing Trace Options . . . 35
Modifying a Trace . . . 37
Switching Log Data Sets . . . 38
Stopping a Trace . . . 38
Purging a Trace . . . 38
Replicating a Trace . . . 38
Quiescing Logging . . . 38

Chapter 4. Managing History Traces . . . 39
The Trace Directory . . . 39
Displaying Historical Trace Data . . . 40
Target Specification . . . 40
Using the History Traces Application (Option 5) . . . 41
Field Definitions . . . 42
Primary Commands . . . 43
Line Commands . . . 45
Batch Maintenance Jobs . . . 48
Trace Log Data Set Jobs . . . 48
Defining a Trace Log Data Set . . . 48
Archiving a Trace Log Data Set . . . 48
Restoring an Archived Trace Log Data Set . . . 48
Printing a Trace Log Data Set . . . 49

Chapter 5. Printing a Trace 51
Overview 51
Types of Reports 51
Summary Traces 51
Detail Traces 52
Detail Traces with I/O Events 52
Primary Input - Trace Log Data Set(s) (TLDS) 53
Printing from Archived TLDSs 53
Secondary Input - SMF or GTF 53
Output Formats 54
Controls 54
JCL 55
Output Format 58
Controls for Reporting 59
Structure of a Report Request 59
Request Keywords 60
Identification Options 61
Selection Options 62
Report Options 65
Special Report Summarization Options for DBIO 69
Formatting Options 70
Default Entry 71
Return Codes 71
Examples of Output 72
Accounting Summary Report 73
Workload Summary by Time 74
Accounting Detail—Long 75
Accounting Detail—Short 77
SQL Statement Summary 78
Thread Detail Events 79
SQL Text and EXPLAIN 80
Lock and EDM Pool Failures 81
I/O Summary by DB/TS 82

Chapter 6. Solving Problems 83

Chapter 1. Requesting a Trace

You can issue requests to

- Start a trace easily through data entry panels on which options can be specified and defaults are displayed. Submitting the request invokes the trace data collection service (ATRAC).
- Move quickly from summary trace displays to detail displays of a specific event.
- Start a trace from an AutoOPERATOR EXEC.
- Start a series of trace requests automatically when the system starts or at your request.
- Log a trace to one or more VSAM data sets for later online recall or batch printing.
- Print a trace display to the TS Image log or your BBISPRNT data set.

These methods are described in the following sections.

Starting a Trace

A trace can be started by:

- Using Option 4 (Current Application Traces) from the Primary Option Menu:
  1. Select Option 4 for a list of current traces.
  2. Type ST (Start Trace) on the COMMAND line, which displays the trace data entry setup panel.
- Defining ATRAC requests in a member of your BBI-SS PAS BBPARM data set (see “Grouping Requests” in Volume 2)
- Writing an EXEC to issue the SET command for the ATRAC service (a BMC Software AutoOPERATOR product must be installed)

  Use the IMFEXEC IMFC command followed by the service name, optional parameters, and an identifier for the target DB2 system:

  ```
  IMFEXEC IMFC SET REQ=ATRAC SUMTRC1 TYPE=SUMMARY ELAP=5.0 TARGET=db2ssid
  ```

Displaying a Trace

To display an active trace of completed transactions:

1. Select the list of current traces using Option 4.
2. Use the S line command to select one of the current traces. A scrollable list of all traced transactions is displayed. From that display you can access all of the more detailed displays as well as those that summarize the information.

   Note: If this trace is also being logged, more data may be available by choosing the HISTORY button to read from the log instead of the online buffers.

To display historical trace data from a trace log data set:

1. Select the list of historical trace data sets using Option 5.
2. Use the S line command to select one of the trace data sets. A scrollable list of all traced transactions is displayed. From that display you can access all of the more detailed displays as well as those that summarize the information.

Printing a Trace

A trace that has been logged to a VSAM data set can be printed by:

- Using Option 5 (History Traces) from the Primary Option Menu:
  1. Select Option 5 for a list of trace log data sets, including both those for current traces as well as earlier history traces that have been saved.
  2. Use the P line command to access a data entry panel to create a print job (under ISPF only). See P(PRINT) on page 45.
- Creating and submitting a batch job, as described in Chapter 5, “Printing a Trace” on page 51.

If the trace was not logged, selected screens can be printed by either:

- Logging a series of trace displays to your TS Image log.
  1. Type Y in the LOG field of the display.
  2. Navigate through the displays you wish to save. All displays are logged until the LOG field is set back to N.
  3. Use BBSAMP member ILOGJCL to print the Image log data set. Instructions to select only a subset of displays are in the *MAINVIEW Administration Guide*, “Selecting and Printing Image Log Records.”
- Logging selected screen images to your TS BBISPRNT data set.
  1. Press the PF4/16 key after the display is presented.
  2. Use BBSAMP member SLOGJCL to print the BBISPRNT data set.

Stopping a Trace

A trace service can be stopped by:

Note: A stopped trace is marked complete and stops collecting trace data, but the data remains available for viewing until the request is purged. A purge both stops data collection and deletes any accumulated data.

- Stopping the trace request with a Z line command

  Use the Z line command from the Current Traces application (Option 4) as described in Chapter 3, “Controlling Current Traces (Option 4)” on page 33.

- Setting a stop time for automatic completion of data collection

  Specify the STOP value (as a timestamp or interval count) on the Start or Modify panel for the trace request. The collected data remains available for viewing until the trace is purged.

  Note: The TRTIME parameter in DMRBEX00 member BBPARM can be used to set a default trace duration that is primed in the Start Trace data entry panel.

- Purging a request from the list of Current Traces with a P line command

  Use the P line command from the Current Traces application (Option 4) as described in Chapter 3, “Controlling Current Traces (Option 4)” on page 33.

- Purging a service request with a SET request

  Issue a PRG request with a SET request from the Service Display panel (Option 1 from the Primary Option Menu), BBPARM (see “Grouping Requests” in Volume 2), or an AutoOPERATOR EXEC (AutoOPERATOR must be installed). For example:

  ```
  SET
  PRG=reqid|ALL
  ```

- Stopping a service request with a SET request

  Use the STOP parameter with the SET request. For example:

  ```
  SET
  REQ=ATRAC,START=11:00:00,STOP=11:10:00
  ```

  At the end of 10 minutes the request stops, but the data remains available for viewing.

  These SET parameters are described in “The SET Timer Request” in Volume 2.

Chapter 2. Collecting Trace Data

DB2 traces are activated and terminated transparently to the user as needed to satisfy trace requests. When MVDB2 first establishes a connection to a target DB2 subsystem (either at BBI-SS PAS startup or DB2 startup), it uses the Instrumentation Facility Interface (IFI) to reserve one OPx destination (where OPx is an internally defined OP1 to OP8 DB2 buffer). The BBI-SS PAS has a DB2 thread, but it is inactive.

Multiple traces can be started with different selection criteria specified with the ATRAC request. A maximum of four detail traces per DB2 subsystem and multiple summary traces (all summary traces share a single OPx destination) can be requested. If four detail traces and 20 summary traces are started for the same DB2 subsystem, five (maximum) OPx destinations are used. DB2 has a a maximum allowable number of eight OPx destinations for the entire DB2 subsystem. The MVDB2 trace maximum limits the number of trace destinations used for the target DB2 by the BBI-SS PAS.

Note: When a summary trace or one or more Workload Monitor services are started, a DB2 trace is shared by all the summary traces and Workload Monitor services. This trace also collects the data for the Lockout History display.

When the first application trace or workload monitor request for a DB2 subsystem is activated, MVDB2 issues the MODIFY TRACE command to request the IFCID (Instrumentation Facility Component Identifier) 3 records. If the DB2 accounting trace level 1 is already active, the existing records are simply passed on by DB2 to this additional destination, causing minimal overhead. If the accounting trace levels 2 and/or 3 are active, the IFCID 3 records passed to MVDB2 contain the corresponding data fields, exactly as documented for the DB2 SMF 101 accounting records.

If the DB2 accounting trace level 1 is not active, DB2 begins collecting data equivalent to this level of tracing to build the IFCID 3 records passed to the IFI. Accounting trace level 1 is **not** shown as active, and no records are written to SMF or GTF.

When all application trace or workload monitor requests are stopped, the MODIFY TRACE command is issued again to remove the request for IFCID 3 which quiesces DB2 IFI processing.

At BBI-SS PAS or DB2 shutdown, MVDB2 issues a STOP TRACE(MON) TNO(x) command through the MVS command interface.

Note: An operator can issue a manual DB2 command to stop the application trace used by MVDB2. If this occurs, any active ATRAC requests are quiesced. To reactivate tracing, all ATRAC requests must be purged. MVDB2 then attempts to reestablish the application trace when the first new trace or monitor request is entered.

Both summary and detail trace entries are identified by a DB2 accounting record and are only made available for display when this record is received. To view detail trace data for a currently active thread, EXPAND from the DUSER display for that thread to UTRAC (see "UTRAC—User Detail Trace" on page 380). The trace entries are often referred to as transactions since there may be more than one accounting record per thread.

Trace Authorization

The trace services are distributed by BMC Software with a security access code of A.

The DB2TRACE keyword authorizes a user to start DB2 traces and to specify the types of trace allowed.

The TRALLOC keyword authorizes a user for dynamic allocation of trace log data sets by the BBI-SS PAS. This keyword is specified in the USERID member of the BBPARM data set.

Users can have their own members with their user IDs as the member names, or the default $USERID member can be used. *Implementing Security for MAINVIEW Products* describes how to use DB2TRACE.

Detail Trace Data Collection

For each detail trace started, a separate DB2 trace is started by the following internal command:

```
-STA TRACE(MON) IFCID(x,x,x...) DEST(OPx)
```

where

IFCID(x)
: Is a set of IFCIDs. The type of detail trace specified with the ATRAC request (TYPE= D|SQL|SCAN|IO|LOCK|DDF|DFFVTAM) determines the set of IFCIDs collected (see Appendix B, “IFCID Table (ATRAC Data Collection)” on page 409).

- D (detail) collects the most important IFCIDs, including all exception events.
- SQL includes all detail events plus SQL statements.
- SCAN includes all detail events plus scan and SQL events.
- IO includes all detail events plus I/O information.
- LOCK includes all detail events plus lock requests.
- DDF includes all detail events plus all DDF events except those related to VTAM.
- DDFVTAM includes all detail events plus VTAM-related DDF events.

Note: Collecting additional types of events incurs more overhead and requires larger trace buffers in MVDB2.

OP(x)
: Is an internally defined DB2 buffer destination (OP1 to OP8).

These traces are displayed with a type of MON by a DB2 -DIS TRACE(*) command and have a destination of OPx.

Detail Trace Online Data Storage

For each active detail trace, there is a set of data collection buffers. The buffer allocated by DB2 is formatted and passed to a data collector function in the BBI-SS PAS. The data collector inserts the formatted variable length data into the trace display buffer associated with each active detail trace. The size of the buffer is specified by the user as described below.

Figure 2 shows how the trace data is collected and stored in the buffers for display by the DTRAC service.

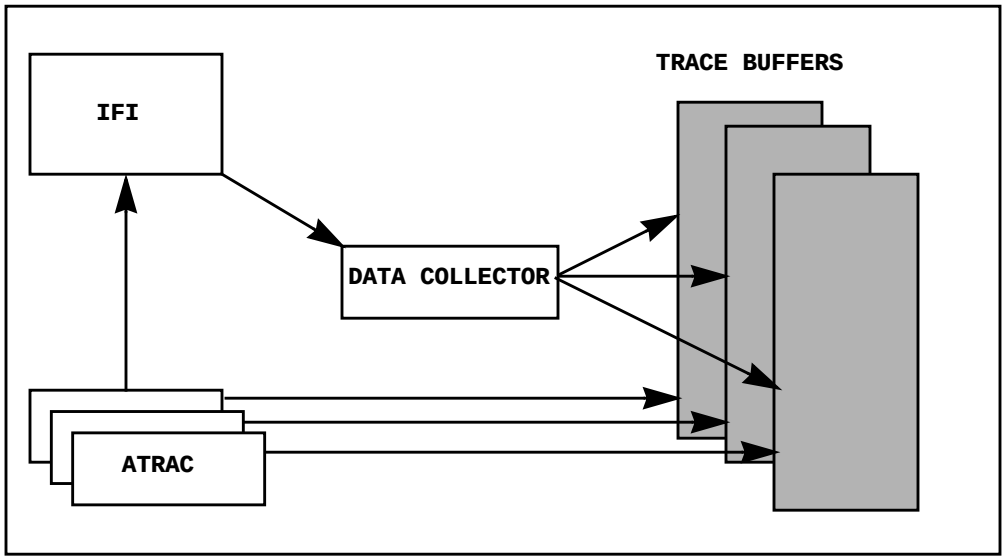

Figure 2. Detail Trace Buffer Storage

There are three kinds of buffers used for a detail trace:

- An internal buffer that is filled by DB2's Instrumentation facility.

  This buffer is managed internally by MVDB2 and DB2 in extended CSA. You cannot specify its characteristics.

- A pool of data collection buffers maintained by MVDB2.

  A pool exists for each active detail trace in the BBI-SS PAS extended private area storage. A buffer is allocated at CREATE THREAD per active thread being traced and more may be needed as the thread progresses. You can specify the size and number of these buffers with the following parameters, which are specified in the DMRBEX00 member of the BBI-SS PAS BBPARM data set (defaults per DB2) or in the data entry panel for the ATRAC request (see Figure 3 on page 14).

  Note: ATRAC TRSIZE and TRBUFF are primed with the TRSIZE and TRBUFF values specified in DMRBEX00. If TRSIZE or TRBUFF is defined with an ATRAC request, their values override any value specified for these keywords in DMRBEX00.

TRBUFF=nnn
: Where nnn is the number of buffers to be allocated.

TRSIZE=nnnK
: Where nnn is the size in K of each buffer. If K is not specified, the value of nnn is in bytes. It is rounded to a multiple of 4K during processing.

For example, DMRBEX00 could specify:

```
TARGET=DB2P
TRBUFF=20
TRSIZE=400K
```

Total pool size is 20*400K = 8,000K

- A single buffer per trace containing displayable data allocated in BBI-SS PAS extended private area with a GETMAIN.

  Its default size is specified in the BBIISP00 member of the BBI-SS PAS BBPARM data set. It can be overridden with the use of the STORAGE parameter specified with the ATRAC request (valid for a summary or detail trace). Entries are wrapped (default of WRAP=YES; new data overlays oldest data) unless you specify WRAP=NO with the ATRAC request. WRAP can be changed at any time by using the M line command from the Current Traces application (Option 4) to modify the ATRAC request.

Note: See "Starting a Detail Trace" on page 21 for detailed recommendations about how to set these values.

Trace Logging

The trace facility allows you to log both summary and detail trace data to external VSAM data sets.

Each trace request is assigned its own data sets. This allows the data from ad hoc application testing traces to be managed separately from those set up by database administrators or systems programmers for system-wide continuous surveillance of the total workload or thread exceptions.

To allow flexible space management, each trace can be logged to a single or multiple data sets, either pre-allocated by the requestor or automatically allocated and managed by the trace facility.

A casual user of the trace can use trace logging very simply by accepting all defaults, while the system administrator has all the options needed for complete automation and control.

Trace logging provides the following flexible reporting capabilities:

- **All data for an active trace can be viewed online**

  If the desired data is no longer available in the online buffers, a HISTORY expand button on the trace displays (LTRAC, STRAC, DTRAC) can be used to retrieve the data from the current log data set where it can be viewed and analyzed with all the online trace displays.

- **Historical trace data can be viewed online at any time**

  All trace log data sets are tracked in an online trace directory. The History Traces application (Option 5) displays these data sets and allows online recall with a simple line command, whether the trace was written the same day, last night, a year ago, or even on another system. This provides these benefits:

  – Data from completed traces is saved in the trace logs so that no data is lost during system interruptions.

  – The older data from an active trace with multiple log data sets can be accessed through this application.

  – When the current log data set for an active trace is accessed through this directory, a CURRENT expand button on the trace displays provides automatic switching to displays of the online buffer data.

  – Online administration of the trace logs simplifies the tracking of which traces are valuable enough to keep for later reporting and which should be deleted after review. An additional option allows the submission of a batch job to print any trace.

  – Traces can be kept as transaction profiles for later comparisons after application or system changes that may affect performance. A trace of an application on the development system can be compared to the same trace on production to easily spot access path changes or performance degradation not shown by EXPLAIN.

- **Short-term workload history is always available**

  The workload summary displays TSUMx and TSTAT can be used online against the data from any of the trace logs to analyze either workload trends or thread exceptions at a particular time or for a specific plan or user. The accounting data is always available to back up these summaries, and for a detail trace the detail events are there too, already correlated for ease of use.

- **Traces can be printed in batch for offline review**

  All trace displays and data levels shown online can also be reproduced as batch reports for hardcopy review. You can choose between a simple application summary report with one line per accounting record, all the data from that accounting record, workload summaries by authorization ID, plan or time, or even the complete detail trace of chronological events with event pop-ups.

- **All summary trace reports can be printed directly from SMF**

  All the displays available for a summary trace are also available as batch reports from the SMF accounting records. If there is no active trace in MAINVIEW for DB2, it is still possible to produce the equivalent reports. These reports can be produced from unloaded SMF data sets, from the current SYS1.MANx data sets, or from a GTF trace.

- **Tracing of very large applications**

  The logging facility also enables tracing of long-running applications without loss of data because of online buffer limitations.

ATRAC—DB2 Application Trace

Data collection must be activated by starting the ATRAC service before data can be displayed with the application trace services. There are several ways to activate an ATRAC request as described in Chapter 1, “Requesting a Trace” on page 3. The usual way is from the ATRAC online data entry panels. These panels can be accessed by typing ST (START TRACE) on the COMMAND line of the list of current traces (Option 4 from the Primary Option Menu).

The online data entry panels present all trace options. Specify the options you want and press ENTER. The activation of any trace request initiates data collection from the specified subsystem. Multiple trace requests with different selection criteria can run concurrently. Thus, you can tailor your trace data for specific needs.

Starting a Trace

When Start Trace (ST) is requested, the initial data entry panel displays a default request for a summary trace (S option for the TYPE parameter), as shown in Figure 3 on page 14. Changing TYPE to D provides the standard detail trace, including SQL.

You also can start a trace with exception filters, detail trace options, and/or trace logging options from an exec or from a BLKxxxxx member in your BBPARM data set. For example, the following exec starts a detail trace that collects SQL, scan, I/O, and lock events in addition to the events collected by the standard detail trace:

```
IMFC SET REQ=ATRAC PLANNAME TARGET=DB2P TYPE=SQL,SCAN,IO,LOCK
STORAGE=4000K WRAP=Y DB2PLAN=planname TITLE=title GROUPSQL=Y
```

In a BBPARM BLKxxxxx member, the same trace can be specified as:

```
REQ=ATRAC PLANNAME TARGET=DB2P TYPE=SQL,SCAN,IO,LOCK
STORAGE=4000K WRAP=Y DB2PLAN=planname TITLE=title GROUPSQL=Y
```

See “SET Keyword Parameter Options” in Volume 2 of the *MAINVIEW for DB2 User Guide* for a complete description of all the keywords you can use.

Start DB2 Trace Request Panel

A trace can be started using only this panel. However, you can access additional data entry panels to override default settings to specify

- Exception filters (to reduce the amount of data saved)
- Detail trace options (to trace additional events)
- Trace logging options (to specify additional data set allocation and processing options)

```
BMC SOFTWARE ---------------  START DB2 TRACE REQUEST -------- PERFORMANCE MGMT
COMMAND ===>                                                   TGT==> DB2D

PARM     ==>         (Trace identifier)     START ==>           (hh:mm:ss)
TYPE     ==> S       (S-Summary,D-Detail)   STOP  ==>           (hh:mm:ss/#min)
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 3. Start DB2 Trace Request Panel

Keyword	Operand/Description
PARM	id

If you want to run multiple trace requests, specify an ID in the PARM field to make the trace unique. However, blank can also be used as an identifier for one trace. The identifier is used to specify which set of trace data is to be displayed with the application trace services.

TYPE S|D

S (the default) requests a summary application trace of completed transactions.

D requests a detail application trace that also collects major events in the life of the transaction.

STORAGE n

Determines the size of the trace display buffer to contain the trace entries per active trace. The size is specified in K format, where K is the number of 1024 byte blocks; for example, 200K is:

```
200*1024=204800 bytes
```

This value is primed from the default specified in DMRBEX00. If no value is specified, the value from BBIISP00 is used.

Note: For a detail trace, this value must be at least four times TRSIZE. If the specification is less than four times TRSIZE, the STORAGE value will be adjusted upward.

LOGTRAC N|Y

N is the default.

Y writes all trace records for this request to a unique trace log data set (TLDS) for this trace. If Y is specified with no additional summary or detail trace logging options (see "Specifying Trace Logging Options" on page 28), a single data set is dynamically allocated using the defaults specified in DMRBEX00 for this DB2.

Note: You must be authorized to use dynamic allocation.

TITLE c...c

Defines a title for the trace (1 to 24 characters). If you are logging the trace, BMC Software recommends that you specify a more specific title to help identify the contents of this trace at a later time.

START hh:mm:ss

Requests trace start time. If the time entered is more than 10 minutes prior to the current time, 24 hours are added to the specified time and the request is started the next day. To start a request at midnight, specify 24:00:00.

Default is the next full minute.

STOP nnn|hh:mm:ss

Requests trace stop limit as a timestamp or number of minutes to process. The default is set from TRTIME in DMRBEX00.

Processing ends at the end of the last interval before the specified stop time. This time is displayed in the STOP field when the request is viewed with the R, P, M, and W line commands from the Current Traces application.

WRAP Y|N

The buffer is used in a wraparound method to store the entries and the wrap can be automatic, WRAP=Y (default), or user-controlled, WRAP=N.

RST Defines the restart option to be used when a service is quiesced because of an inactive DB2 subsystem or SETBLK=RRR request. Default is HOT.

HOT Restarts the trace automatically without the loss of previous data.

PUR Purges the trace automatically when the target DB2 starts.

QIS Keeps the trace in a quiesced state until it is purged by an authorized user.

Selection Criteria:

A list of operands can be specified in each of the selection criteria fields; for example:

```
DB2CONN ===> IMSA,IMSB
```

Note: For the selection to match a trace record, selection criteria keywords are connected with an AND operation, and the operands within a keyword are connected with an OR operation.

Note: For a detail trace, you must specify at least one valid plan or authid.

DB2PLAN name

Qualifies a workload trace by a one- to eight-character DB2 plan name.

Summary Trace
The maximum number of operands is the total field length (60 characters). The + character can be used as a name qualifier (see "Qualifying Requests" in *Using MAINVIEW*).

Detail Trace
You can specify up to eight plans to be traced. If more than one DB2AUTH operand is specified, only one plan name can be specified.

Note: The + and * characters cannot be used as generic operand characters for detail trace.

DB2AUTH id

Qualifies a workload trace by DB2 authorization ID.

Summary Trace
The maximum number of operands is the total field length (60 characters). The + character can be used as a name qualifier (see "Qualifying Requests" in *Using MAINVIEW*).

Detail Trace
The maximum number of operands is eight. If more than one DB2PLAN operand is specified, only one authorization ID can be specified.

Note: The + and * characters cannot be used as generic operand characters for a detail trace. The only exception is the use of a single + character as the operand for DB2AUTH, which traces all DB2 workloads and can cause excessive overhead except in a lightly-loaded test system. Therefore, this option is subject to security restrictions.

DB2CONN name

Qualifies a summary workload trace request by a one- to eight-character connection name.

Note: For a detail trace, this parameter does **not** reduce data collection. Non-matching records are filtered out after they are received from the DB2 trace.

The maximum number of operands is the total field length (60 characters). The + character can be used as a name qualifier (see "Qualifying Requests" in *Using MAINVIEW*).

DB2CORR id

Qualifies a summary workload trace request by a 1- to 12-character correlation ID.

Note: For a detail trace, this parameter does **not** reduce data collection. Non-matching records are filtered out after they are received from the DB2 trace.

The maximum number of operands is the total field length (60 characters). The + character can be used as a name qualifier (see "Qualifying Requests" in *Using MAINVIEW*).

DB2LOC name

Qualifies a summary workload trace request by a 1- to 16-character remote location name. Both server and requester activity with this location is selected (DBAT and DIST threads).

Specify an IP address in quotation marks.

Note: For a detail trace, this parameter does **not** reduce data collection. Non-matching records are filtered out after they are received from the DB2 trace.

The maximum number of operands is the total field length (60 characters). The + character can be used as a name qualifier (see "Qualifying Requests" in *Using MAINVIEW*).

DB2PKG name

Qualifies a summary or detail workload trace request by a 1- to 8-character DB2 package name (only when accounting class 7 or 8 is active).

The maximum number of operands is the total field length (60 characters). The + character can be used as a name qualifier for a summary trace (see "Qualifying Requests" in *Using MAINVIEW*).

CONNTYPE TSO|IMS|CICS|BATCH|CAF|IMSMPP|IMSBMP|IMSTBMP|IMSCTL|DLI|SYSSERV|APLSERV|UTIL|blank

Qualifies a summary or detail workload trace by connection type.

Blank collects data for all connection types.

Additional Trace Options Processing

You can access additional data entry panels to override default settings to specify exception filters, detail trace options, or trace log data set options by specifying Y for one or more of the additional trace options.

When you select Y for an option, you are presented with an additional menu. Once you are finished with that screen, press the ENTER key to process. If there are more extension screens to be processed, you are presented with the next screen without returning to the Start DB2 Trace Request panel.

When all extension screens are processed, the Start DB2 Trace Request panel displays an asterisk (*) next to each processed option. The request is not submitted until all the options display either N or *. This gives you the opportunity to review and change your options before sending the request. For example:

- If you change an * to an N, the items specified on that extension menu are dropped.
- If you change an * to a Y, that extension menu is displayed primed for update with the values previously specified.
- If you change an N to a Y, that extension menu is displayed.

Note: Once a trace is activated, only certain options can be changed. To change the others, the trace must be stopped and reactivated. See "Modifying a Trace" on page 37.

Selecting Trace Data for Retention

The Exception Filters are keywords that retain only the required data in the online trace buffer or the trace log data set (TLDS). They can be used to

- Detect occurrences of specific exception conditions
- Limit the amount of buffer or DASD storage required by retaining only the most interesting transactions for analysis

If you specified Y in the Exception Filters field on the Start DB2 Request panel, the data entry panel shown in Figure 4 is displayed.

Note: Data that causes a trace entry to be filtered is highlighted in the STRAC (Summary Trace Entry) display (see "STRAC—Summary Trace Entry" on page 185).

```
BMC SOFTWARE ---------- DB2 TRACE EXCEPTION FILTERS ----- PERFORMANCE MGMT
COMMAND ===>                                                 TGT -- DB2D

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

Figure 4. DB2 Trace Exception Filters Panel

Exception Filter Keyword Descriptions

The keywords and their values are described below. The keyword values are user-defined. A < prefix can be used with a numeric value to define a less-than condition. If a numeric value is not defined as a less-than condition, it is compared to a greater-than or equal-to condition. The specified keywords are ORed; a trace record is retained if any one of the comparison conditions is valid.

For example:

```
ELAP   => 5.5
```

Collects only records with an elapsed time equal to or greater than 5.5 seconds.

```
GETRIO => <5
```

Collects only records that have a GETPAGE to READ I/O ratio less than 5.

Note: For trace exceptions requiring AND logic, you can give the trace a special name (in the PARM field) to activate this logic. The name must be in the format N#xxxxxx, where xxxxxx can be one to six alphanumeric characters.

Table 1 is an alphabetical listing of the keywords with a description of each value that can be used.

Table 1. Exception Filter Keyword Descriptions

Exception Filter Keyword	**Description**
ABORT	Y selects aborted transactions.
CPU	CPU transaction time. The time can be either in hours, minutes, and seconds as hh:mm:ss or in seconds as ss (maximum of three decimals).
ELAP	Elapsed time of transaction. The time can be either in hours, minutes, and seconds as hh:mm:ss or in seconds as ss (maximum of three decimals).
Note: The following keywords must be numeric. They can be specified in K; for example, 100K.	
CLAIMDR	Number of claim / drain failures (QTXACLUN + QTXADRUN).
GETPAGE	Number of GETPAGE requests.
GETRIO	Ratio: Number of GETPAGE requests issued per READ I/O. **Note:** A < prefix should be used to define this value. The IBM *DB2 Administration Guide, Volume III*, suggests that a ratio below 1.5 can initiate poor read efficiency.
HPFAILS	Number of hiperpool read or write failures (QBACHRF + QBACHWF).
INCRBIND	Number of incremental BINDs issued.
LOCKESCL	Number of lock escalations.
LOCKSUSP	Number of lock suspensions.
LOCKTBL	Number of LOCK TABLE requests.

Table 1. Exception Filter Keyword Descriptions (Continued)

Exception Filter Keyword	Description
MAXLOCK	Maximum number of locks held.
PFREADS	Number of prefetch read I/Os (QBACSIO).
PFREQS	Number of prefetch requests (QBACSEQ).
PGUPD	Number of page update requests.
PRLLFALB	Number of parallel groups that fall back to sequential processing (QXDEGCUR + QXDEGESA + QXDEGBUF + QXDEGENC).
PRLLGRP	Number of parallel groups executed (QXTOTGRP).
PRLLRED	Number of parallel groups executed at a reduced degree (QXREDGRP).
PWAITIO	Percent of elapsed time for I/O waits (QWACAWTI/QWACASC).
PWAITLK	Percent of elapsed time for locks (QWACAWTL/QWACASC).
PWAITOT	Percent of elapsed time for other waits ((QWACAWTE+QWACALOG)/QWACASC).
PWAITPF	Percent of elapsed time for waiting for prefetch to complete (QWACAWTR/QWACASC).
READIO	Number of READ I/Os issued.
RIDFAIL	Number of RID pool failures (QXNSMIAP and/or QXMRMIAP).
RLF	Y selects transactions that were governed by RLF.
SQLCTL	Number of SQL control statements, such as GRANT or REVOKE.
SQLDDL	Number of SQL Data Definition Language statements.
SQLDYN	Number of SQL PREPARE statements.
SQLFETCH	Number of SQL FETCH statements.
SQLSEL	Number of SQL SELECT statements.
SQLTOT	Total number of SQL statements.
SQLUID	Number of SQL UPDATE, INSERT, and DELETE statements.
TIMEOUT	Number of timeout/deadlocks (QTXATIM).
UIDCOM	Ratio: Number of SQL UPDATE, INSERT, and DELETE statements issued per COMMIT.

Starting a Detail Trace

To start a detail trace, specify D in the TYPE field on the Start DB2 Trace Request panel shown in Figure 3 on page 14.

A maximum of four detail traces can be activated. An ID must be specified as a parameter to make each request unique. Tracing begins immediately even for currently active threads. If a transaction is already executing, the data for events that occurred before the trace started will be missing.

See "Recommendations When Starting a Detail Trace" on page 25 for points you should consider before starting a detail trace.

Note: You must be authorized to make detail trace requests.

Detail Trace Options

To change the default options, access the following extension data entry panel by specifying Y in the Detail Trace Options field on the Start DB2 Trace Request panel shown in Figure 3 on page 14. After changing this panel, press the ENTER key to process the options.

```
BMC SOFTWARE --------------- DETAIL TRACE OPTIONS ------------ PERFORMANCE MGMT
COMMAND ===>                                                   TGT -- DB2D

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 5. Detail Trace Options Panel

Additional Events:

A default application trace request collects major events in the life of the transaction, including all exception events and SQL statements.

A basic detail trace (not including SQL) collects accounting summary (SMF 101) events plus the following events:

- Lock summary
- TERM THREAD
- Utility phases
- PLAN BIND, REBIND, and FREE
- EDM requests
- SORT
- Plan allocation
- TABLESPACE OPEN and CLOSE
- Package allocation
- Authorization failure
- Lock suspension
- Multiple Index Access Path failures
- ABORT
- COMMIT
- CREATE THREAD
- Lock time-out and deadlock

By specifying Y on the Detail Trace Options panel, you can request that the following additional types of data be collected:

- SQL (default = YES)

  Traces all SQL statements and produces a *mini-explain* for dynamic SQL. Elapsed time, CPU time, return code, number of pages scanned, and number of rows scanned by each SQL statement are reported.

- SCANS (includes SQL)

  Traces all types of scans within the context of SQL statement execution. Number of pages scanned and number of rows scanned by table space and/or index are reported.

- DDF

  Reports all DDF events up to the VTAM boundary. This includes the RDS/DRDS interface, AUTHID translation, and distributed SQL.

In most circumstances, this is the most you would set up to trace. At this level you generally have enough information to analyze SQL and table access patterns or to identify the origin of a problem, while the overhead is still acceptable in a test environment.

Adding I/O events also can be valuable when physical data access by table space and index space is a concern.

- I/O

  Traces all I/O activity on table spaces and indexes, both synchronous and asynchronous. Elapsed time and pages read or written are reported.

Only in very rare situations do you need to know exactly what is going on at the lock request and VTAM levels to identify a problem. The overhead incurred in tracing the following events can be quite high:

- LOCKS

  Traces every request to IRLM. Lock token, request type, and return code are reported.

  Note: Even without tracing lock events, a lock summary is available in DTRAC (see "Lock Summary Pop-Up (IFCID 020)" on page 318).

- DDFVTAM

  Traces VTAM activity resulting from DDF requests. Events such as VTAM macro and send/receive buffers are reported.

See Appendix B, "IFCID Table (ATRAC Data Collection)" on page 409 for a list of all IFCIDs in these groups.

Event Compression:

GROUP SQL Y|N

Identical SQL events are grouped into one to reduce storage usage.

Data Collection Buffer Options:

TRSIZE nnnK

Size of the trace collection buffer to use. You may need to increase it to trace a long-running application.

This number is rounded to a multiple of 4K. This buffer is allocated to individual threads to receive trace data from DB2. At buffer overflow or term-thread time, the content of the buffer is transferred to the display buffer (defined by the STORAGE parameter) and to the trace log data set if logging is requested.

The default is obtained from member DMRBEX00 of the BBPARM data set. It is recommended you start with 400K and do not reduce it below 100K.

Note: Storage constraints dictate that the size of the trace buffers must be balanced with the number of trace buffers. A given transaction (thread) can use up to two buffers at one time. Thus, if the trace is not logged and the second trace buffer overflows, you may not have the complete trace for the transaction traced. (Therefore, logging is recommended.) On the other hand, fewer very large trace buffers can cause you to lose data completely simply because there are not enough trace buffers to go around for all the concurrent threads.

TRBUFF nnn

Number of trace buffers to allocate for a trace request. This determines how many concurrent threads can be traced.

TRBUFF and TRSIZE together define the total capacity for receiving trace data from DB2. (STORAGE defines the real time display capacity.)

The number of trace buffers should be at least double the number of concurrent active threads being traced in DB2 plus 20%. Preferably, it should triple that number. The extra capacity is needed to handle buffer overflow and the turnaround delay in unloading data into the display buffer.

The default is obtained from member DMRBEX00 of the BBPARM data set.

For example, to trace a long-running application, qualify the request so that only one thread is traced at a time. Set TRBUFF to 3 and increase TRSIZE to hold more events.

If you specify logging, each filled buffer is written to VSAM. You can use online recall from the trace log (History Trace) to see the complete thread activity. TRSIZE can therefore be kept smaller.

Conversely, if you are tracing online transactions, you will probably need to set TRBUFF higher to capture multiple concurrent threads, but the number of events (and TRSIZE) is smaller.

Recommendations When Starting a Detail Trace

This section discusses several points you should consider before starting a detail trace.

Choosing Which Events to Trace

The default detail trace includes many important special events in the life of a thread (for example, Create Thread, Plan/Package Allocation, Sort), as well as all SQL statement events. The majority of application problems can be resolved with this data, so most detail traces should be at the SQL level to limit trace overhead and storage usage.

- Add SCANS and/or I/O when activity per table space / index space is required.
- Add DDF events for distributed threads.
- Add LOCKS or DDFVTAM only for very special situations.

  Both of these trace types add a large number of events and overhead to a trace request. Review the events captured by these two types and make sure that they really give you the information you need before activating them. See "Analyzing Lock Usage Without Tracing LOCK Events" on page 26 for special considerations for lock analysis.

Specifying GROUPSQL=Y can reduce storage requirements by grouping identical SQL events, such as a series of FETCHes.

Specifying Selection Criteria

Limit the number of concurrent threads to be traced using DB2AUTH or DB2PLAN. Exception filters reduce the amount of data saved, but do not affect the amount of data collected from DB2.

Reviewing Storage Requirements

To collect detail trace data from DB2, MAINVIEW for DB2 (MVDB2) uses several buffers for different purposes. The MVDB2 administrator sets up defaults per DB2 that are sufficient for most trace requests. However, if you are tracing more events or more concurrent threads, you may need to increase these defaults. Otherwise you may receive various messages that detail data (some events) has been lost. The STORAGE parameter (first panel, needed for both summary and detail traces) determines how much you can view from online storage (Current Traces). TRBUFF and TRSIZE (Detail Trace Options panel) control the detail event capture buffers.

- If the number of concurrent threads to be traced is not limited to one, ensure that TRBUFF is two to three times the number of threads. For example, if you are tracing a specific plan, but three users may be executing it at the same time, set TRBUFF to 8–9.
- If you are tracing long-running threads, or have specified more event types, review the default TRSIZE to consider increasing the buffer size to hold more events. Often in this case you can limit the number of concurrent threads to one and reduce TRBUFF to three to five. (Three buffers may not be enough to trace a single thread when running on a fast CPU and logging many events.)
- A single cylinder trace log data set holds approximately 700K of data. If TRSIZE is greater than 700K, it requires more than a one-cylinder trace log data set to hold a full trace buffer.

- Consider logging the trace, since without logging you are limited to viewing only the data from the last two filled buffers per thread, and only what is currently in the STORAGE buffer. With logging, multiple buffers can be written per thread and combined automatically when recalled online or printed. It is preferable for this type of trace to allocate one large trace log data set (TRCYLS), since some data can be lost if events for one thread span two data sets. You can view data online from one trace log data set at a time, but can print full reports from multiple data sets.

  If you have logged a trace, it is preferable to view the trace data from History Traces (the logs) instead of Current Traces for two reasons:

  – When the tracing is complete, you can purge the trace request from Current Traces and free up the storage; storage is then allocated only when you are actually viewing the logged trace data.

  – The trace data for long-running applications may be more complete on the logs (multiple combined buffers) than in the online STORAGE buffer.

Tracing Long-Running Threads

Sometimes, no data is shown in an active trace for a long-running thread.

The data is being collected, but is usually only shown in LTRAC and STRAC when the thread completes and the DB2 accounting record is available. While the thread is active, you can view the events with the UTRAC display, which is like the DTRAC display but is accessible from DUSER for that thread.

You also can set up trace logging for a long-running thread, specifying one or more small trace logs.

Each time a trace log fills up, a dummy accounting record is written for any threads being traced that are still incomplete. These entries then show up on LTRAC (with a status of INCMP) and can be selected for viewing. The STRAC display obviously has none of the accounting data, but does have the detail summaries of events (SQL, IO/LOCK, and so forth), and from STRAC you can expand to the DTRAC entries.

If the trace is not stopped, the final detail events and the accounting record information appear in the trace log active when the thread completes, as before.

Specify log wrap YES or NO, depending on which part of the thread you want to keep. With YES, the earlier events can be overwritten.

Analyzing Lock Usage Without Tracing LOCK Events

The first place to look is the Lockout History display that shows you a list of the last 100 timeouts and deadlocks that have occurred. The complete details from these events are available to identify both holder and waiter threads as well as the resource(s) involved.

To analyze locking problems for an application, first look at the accounting record (STRAC) for the counts on maximum page locks, suspensions, lock requests, and timeout/deadlocks. This data is already available with a summary trace. This should help you identify which applications are having—or causing—problems. For further information, a detail trace with SQL events is often sufficient. You usually will want to include SQL so you can more easily determine which statement is causing the problem, but a great deal of information is available with just the basic detail trace.

If you are still searching for the culprit application, you may want to run a basic detail trace with no further events (not even SQL), but covering all the suspected applications. This type of trace causes much less overhead. You could also use exception filters in the trace request to identify threads with lock suspensions, escalations, lock table requests, or those with a high maximum number of locks held.

The following events are available in DTRAC for any detail trace. Analyze this information before deciding to trace LOCK events.

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 TIMEOUT event (display LEVEL=2) that gives complete details.

Note: LOCK-TMO is not available for DB2 4.1.

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.

Another source of information is the DB2 timeout and deadlock messages in the Journal Log or MVS console. If the DB2 messages are being written to the Journal, go to Option L and issue a FIND DSNT37 to find any deadlock (DSNT375) or timeout (DSNT376) messages. They are followed by DSNT501I to identify the resource and lock owner that caused the contention. If you have AutoOperator Solutions installed, you can use the DB2DLOK solution to summarize and analyze the deadlock/timeout messages and LOCKD contention snapshots. Refer to the *MAINVIEW Solutions Guide* for details.

Specifying Trace Logging Options

To specify additional trace logging options, access the following extension data entry panel by specifying Y in the Trace Log Data Set Options field on the Start DB2 Trace Request panel shown in Figure 3 on page 14. After filling out this panel, press the ENTER key to process the options.

```
BMC SOFTWARE -------------- TRACE LOG DATA SET OPTIONS ------ PERFORMANCE MGMT
COMMAND ===>                                                  TGT -- DB2D

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 6. Trace Logging Specification Panel

where

Number of Logs — The number of trace log data sets to be allocated. Specify more than one for continuous traces. This allows automatic switching when a log data set is full.

First Log DSN

Name of the first log data set to be allocated.

If left blank, a default name is constructed as &TRPREFIX.&DB2ID.&TRACEID.mmmdd.Thhmm.V01 where

- &TRPREFIX is your user ID or the prefix specified in DMRBEX00.
- &DB2ID is the target DB2 subsystem ID.
- &TRACEID is the identifier specified in the PARM field on the first Start DB2 Trace Request panel.
- The mmmdd is the current month and day.
- Thhmm is the current time.
- V01 is the required suffix.

If quotation marks are omitted for the log data set name, the specified value is prefixed with TRPREFIX as specified in DMRBEX00. If TRPREFIX is not specified, your user ID is used for the prefix. If quotation marks surround the name, it is allocated exactly as specified.

For example, if there is no TRPREFIX specification in DMRBEX00, you can specify:

```
TEST.V01
```

to allocate userid.TEST.V01.

You can specify up to an eight-character suffix for the data component (see the Data DSN Suffix description on page 30). However, if you want a name totally different from the cluster data set name, you must preallocate the data set.

Overwrite logs

Y|N

Indicates the action to be taken when a log data set is about to be used for trace data without having been reset. Y means that existing data can be overwritten; N means that logging should be quiesced.

Note: This field has no effect if the data set has been reset.

Archive PROC

Name of the entry in PROCLIB to archive a log data set when it is full, logging is switched to the next data set, or the trace completes normally. (See the sample archive job in BBSAMP member DZTBARC; see "Archiving a Trace Log Data Set" on page 48.)

Log switch time

hh:mm

Specifies the time an automatic log switch from the current log to the next log will occur. Valid only if Number of Logs > 1.

Note: If another log is not available at that time (status of EMPTY or RESET), the switch cannot be made.

Disposition

OLD|NEW

OLD indicates all the data sets exist and are valid trace data sets. (See the sample job for allocating a trace log data set in BBSAMP member JXT011; see "Defining a Trace Log Data Set" on page 48.) Existing data is overlaid if Overwrite Logs = Y.

NEW (default) indicates none of the data sets exist. All the data sets are allocated when the trace request is processed. If any of the data sets cannot be allocated, the trace request fails and the data sets successfully allocated are deleted.

Note: If your user authorization specifies TRALLOC=NO, only OLD is accepted.

Volumes

Volume(s) to use when allocating a trace log data set. You can specify up to seven volumes. The default value specified in DMRBEX00 with the TRVOLS keyword is SYSDA.

When the number of logs is one, the data set is allocated on the first volume. When the number of logs is greater than one, the data sets are allocated alternately on each specified volume (if there is enough space) until all data sets are allocated.

Note: A volume specification replaces any SMS class specification. However, if your data center is SMS-controlled, the volume specified may be overridden by your SMS criteria.

SMS-controlled sites do not need to specify volume(s). Non-SMS-controlled sites must specify volume(s) to avoid an IDCAMS-defined cluster error.

Primary CYLS

Primary allocation in CYLS for trace log data sets. The default value specified in DMRBEX00 with the TRCYL keyword is 3.

Note: Allocating a very large data set (with several hundred cylinders) can cause a delay due to formatting by VSAM. In this case, specifying a pre-allocated and pre-formatted log data set is preferable.

Data DSN Suffix

Suffix to be appended to the cluster DSN to make the DSN of the data component. The default value specified in DMRBEX00 with the TRSUFFIX keyword is D.

SMS Storage Class

Name of the SMS Storage Class to be used when allocating this trace data set. The default is specified in DMRBEX00 with the TRSMSSCL keyword.

SMS Data Class

Name of the SMS Data Class to be used when allocating this trace data set. The default is specified in DMRBEX00 with the TRSMSDCL keyword.

SMS Management Class

Name of the SMS Management Class to be used when allocating this trace data set. The default is specified in DMRBEX00 with the TRSMSMCL keyword.

Log Switching Considerations

You can assign a set of logs for a each trace when submitting the trace request. Log switching to the next available data set occurs when

- A log data set is full
- At a time, such as midnight, specified with the trace request
- You request an immediate log switch with the I command on the Current Traces display

If an archive started task was specified on the trace request, that STC is run. It can

- REPRO a copy of the data to tape or another disk data set
- Execute a step to flag the data set for reuse
- Submit a job to print reports

If no data set is available and overwriting logs was specified on the trace request, logging switches with the oldest trace data set.

Replicating a Trace Request

Once you have activated a trace request from the ATRAC data entry panels, you can easily submit another request with similar options. To replicate an existing request:

1. Access the Current Traces application (Option 4).
2. Use the R line command to select an active trace that is similar to the one you want to activate.
3. Specify a unique identifier in the PARM field.
4. Modify as needed any values that are prefixed with an ===>.
5. Submit the new request.

Chapter 3. Controlling Current Traces (Option 4)

You can control active traces by accessing the Current Traces application (Option 4).

```
BMC SOFTWARE ------------------- CURRENT TRACES -------------  PERFORMANCE MGMT
COMMAND ===>                                                   TGT ===> DB2D
                                     INPUT    INTVL ==> 3      TIME --  14:31:36
COMMANDS: ST (START APPLICATION TRACE), 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     CIR11    DB2D    SUM  WKLD  ACTV  ACT
    DETAIL   TRACE OF APDT001T          PWW1     DB2D    DET  WKLD  ACTV
    BIGELAP  ELAPSED OVER 10 SEC        PWW1     DB2D    SUM  WKLD  ACTV
    ***************************** END OF REQUESTS ***************************
```

Figure 7. Current Traces Application

This application is similar to the Active Timer Requests application (see “List Active Timer Requests (Option 3)” in Volume 2), except the list of services shows only current trace requests. This chapter discusses the uses of this application that apply only to current traces.

Field Name	Description
LC	The line command input column. One-character line commands can be typed in this column to view, modify, or replicate the options for a selected request; purge a request; or display HELP information about the service (see “Line Commands” on page 35).
PARM	This field shows the parameters that were defined for the active requests. For a trace, this is the identifier specified when the trace was started to make the request unique.
TITLE	The service title.
USER ID	The logon identification of the user who started the trace.
TARGET	The DB2 subsystem defined as the target of the trace.
TYPE	The type of trace request. The possible values are: SUM Summary trace DET Detail trace I/O Detail trace that includes I/O events
AREA	The DB2 resource area being analyzed. This field could contain: **Field Data** **Description** WKLD DB2 workload

STAT — The service request status which could be:

Field Data	Description
ACTV	The request is active.
COMP	The request executed and completed normally.
HELD	The request is being held and is pending release.
INIT	The request is being invoked for the first time (a start time was specified, but it has not been reached).
INV	The request terminated because of an invalid parameter or measurement. The BBI-SS PAS Journal log contains a descriptive message of the error.
LOCK	A LOCK command was issued for the service or the service abended.
QIS	The service is quiesced because the target DB2 is not active.
RST	The target DB2 restarted. The request is waiting until the current interval expires before performing restart processing as specified by the RST keyword in the original request.

LOG — The status of logging for this trace request. The possible values are

ACT	Logging is active.
QIS	Logging is quiesced.
SUSP	Logging is suspended because the trace is quiesced. Logging resumes when the trace restarts.
blank	Logging was not requested for this trace.

Primary Commands

The following primary commands can be typed on the COMMAND line of the Current Traces application:

- The ST command accesses the Start DB2 Trace Request panel and each of the additional ATRAC options panels. Use this command when you want to start a trace. (See "ATRAC—DB2 Application Trace" on page 13.)
- The HT command accesses the History Traces application. Use this command to access and control the current and historical trace logs. (See Chapter 4, "Managing History Traces" on page 39.)
- The TYPE command can be used to list only the traces of the specified type. The possible types that can be specified are shown in the TYPE column. For example, to list only the detail traces (including the I/O traces), type on the COMMAND line:

  ```
  TYPE DET
  ```

  Type TYPE to return to the list of all the traces.

Line Commands

From this list of current traces you also can use line commands to

- Display the collected trace data
- Show the trace options
- Modify a trace
- Switch log data sets
- Stop a trace
- Purge a trace
- Replicate a trace
- Quiesce logging

Displaying Trace Entries

To access a scrollable list of traced transactions held in the online storage buffer, use line command S (SELECT). The LTRAC display for the selected trace is shown. From here, all other trace displays can be accessed with EXPAND. (See Chapter 9, "Application Trace Displays" on page 109 and "LTRAC—DB2 Trace Entries" on page 115.)

Showing Trace Options

To view the options specified for a trace, use the W (SHOW) line command.

Note: The show request can be initiated from either the Current Traces application (Option 4) or the History Traces application (Option 5).

The Start DB2 Trace Request panel and each of the additional ATRAC options panels are shown for the selected trace. (See "ATRAC—DB2 Application Trace" on page 13.)

Each of the panels show the options specified when the trace request was submitted. However, the Show Trace Logging Specification panel has additional fields that are obtained by reading the current trace log data set.

```
BMC SOFTWARE ---------- SHOW TRACE LOGGING SPECIFICATION ---- PERFORMANCE MGMT
COMMAND ===>                                                 TGT -- DB2D

Number of logs:       3
Log DSN:              CIR1.DB2D.SUMMARY.MAR19.T1125.V01

Recording Start:      19MAR01 11:25:00   Percent used: 100   Status: RESET
Recording Stop:                          Data lost:    No

Overwrite logs:       Y    (Y/N) (Action when all logs used)
Archive STC:          ARCHJCL1   (Blank for none)
Log switch time:      24:00:00   (HH:MM that a log switch is requested)

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

 Volumes:             BAB003,BAB303,BAB009
 Primary CYLS:        3                SMS Storage Class:
 Data DSN suffix:     DATA             SMS Data Class:
                                       SMS Management Class:
```

Figure 8. Show Trace Logging Specification Panel

The Log DSN field is changed to be either the DSN of the current log data set if the trace is active or the last data set used if the trace is complete.

The additional fields are

Recording Start: Date and time this data set was activated for logging.

Recording Stop: Date and time of the last record written to this data set.

Percent Used: Number of pages used divided by the number of pages allocated, expressed as a percent, for this data set.

Data Lost: Indicates (YES/NO) whether any data was lost while writing to this log data set.

Status: The status of the log data set. Valid statuses are

EMPTY	The data set contains no trace records.
UPDATE	The data set is being updated.
USED	The data set contains valid trace data.
RESET	The data set contains valid trace data and has been flagged for reuse.
INCOMP	The data set was not successfully closed (or archived) after an update; for example, in a system failure.

Modifying a Trace

Use the M (MODIFY) line command to modify some of the options for an active trace.

The Start DB2 Trace Request panel and each of the additional ATRAC options panels are shown for the selected trace. (See "ATRAC—DB2 Application Trace" on page 13.)

All values prefixed with an ===> can be modified.

Note: You cannot modify the LOGTRAC option after the trace has started. Thus, you cannot start or stop logging in the middle of a trace with a Modify command.

The following table describes several modifiable options you might want to change after starting a trace where logging is already active:

Table 2. Modifiable Logging Options

Panel Option Name/SET Keyword	Possible Use
Overwrite logs/TRREUSE	Changing from Y to N prevents overlaying data that has not been archived. Changing from N to Y assures that a data set is always available for new trace data, but old trace data may be overlaid before it can be archived.
Log switch time/TRSWTIME	Allows you to change the time when a log switch occurs. For instance you might want to add a data set by increasing TRNUMDS and then force a switch to free up the current data set.
Archive STC/TRARCSTC	To change the name of the archive job, perhaps to archive to a different set of archive data sets.
Number of logs/TRNUMDS	To increase or decrease the number of log data sets. If TRNUMDS is increased and TRDISP=NEW, the additional data sets are allocated before the modify request is considered successful.
Volumes/TRVOLS, Primary CYLS/TRCYL, SMS Storage Class/TRSMSSCL, SMS Management Class/TRSMSMCL, or SMS Data Class/TRSMSDCL	If TRDISP=NEW and you are adding log data sets by increasing TRNUMDS, you may want to change the allocation parameters for these new data sets to cause them to be allocated on different volumes, or to increase their size.

Note: If the number of logs is increased using the Modify command, the additional log data sets must be successfully allocated before the request is considered successful. If they cannot all be allocated, the request fails and any additional data sets that were dynamically allocated (DISP=NEW) are deleted. Logging continues with the original number of log data sets.

Switching Log Data Sets

To switch from the current log data set to the next available one, use line command I (SWITCH).

An archive request for the log data set just inactivated is started if one was specified on the trace request. If the switch command is issued and only one log data set is defined, then logging either reuses the same data set (TRREUSE=Y) or quiesces (TRREUSE=N).

Note: To issue the I (SWITCH) line command for traces other than your own, you must have the authority to purge any request. This authority is specified with the security resource name PMACC#.

Stopping a Trace

To stop an active trace, use line command Z (STOP).

Note: To issue the Z (STOP) command for traces other than your own, you must have the authority to purge any request. This authority is specified with the security resource name PMACC#.

Purging a Trace

To purge an active trace, use line command P (PURGE).

Note: To issue the P (PURGE) line command for traces other than your own, you must have the authority to purge any request. This authority is specified with the security resource name PMACC#.

Replicating a Trace

To replicate a trace, use line command R (REPLICATE).

The Start DB2 Trace Request panel and each of the additional ATRAC options panels are shown for the selected trace. (See "ATRAC—DB2 Application Trace" on page 13.)

Each of the panels show the options specified when the original trace request was submitted. All values prefixed with an ===> can be modified.

Quiescing Logging

To quiesce logging for a trace, use line command Q (QUIESCE LOGGING).

Once quiesced, logging cannot be restarted. The trace continues to run, but no data is written to the log. To restart logging, you must purge and reset the trace.

Note: To issue the Q (QUIESCE LOGGING) line command for traces other than your own, you must have the authority to purge any request. This authority is specified with the security resource name PMACC#.

Chapter 4. Managing History Traces

The History Traces application allows you to access and control the current and historical trace logs for all DB2 systems managed by a BBI-SS PAS. This application can be accessed by

- Option 5 from the Primary Option Menu
- The HT application transfer command from any COMMAND ===> line

It provides

- Online recall of trace data
- Review of trace options
- Online administration of the data sets (purge, free, verify, reset, archive)
- Registering of trace log data sets created on other systems
- Printing of trace reports from the log data sets

The Trace Directory

The trace directory is a VSAM data set that contains an entry for every trace log data set that is known to the BBI-SS PAS. The DSN of the directory must be specified in the parmlib member BBIISP00 with the keyword TRDIR=. There is no default. If specified, it is dynamically allocated at BBI-SS PAS start. Trace logging can occur only when a trace directory is successfully allocated.

Trace log data sets are tracked as long as the data sets exist, but entries can be deleted or added to the directory to allow logs to be restored after archiving or moved between systems.

There is no automatic updating of the directory from events outside the BBI-SS PAS system. There is a batch utility, JXT003, provided in BBSAMP to update your directory for data sets that have been deleted or archived.

Displaying Historical Trace Data

The History Traces application tracks current and historical trace log data sets and allows any of them to be selected and loaded. You are then presented with the standard LTRAC display and can navigate through the various displays in the same way as for a current trace. The message in line 4 specifies ENTRIES IN DATASET instead of ENTRIES IN BUFFER.

In addition, for active or complete traces that have not yet been purged, it is possible to move between the display of active trace data in the buffers to the display of trace history data that has already been written to the current trace log data set.

This is done with a select button on the EXPAND line. This option is available on LTRAC, STRAC and DTRAC, but not on the DTRAC pop-ups.

- The button displays HISTORY if the display is from a current trace with logging. You can view the history trace data set for that trace by selecting the HISTORY button.
- The button displays CURRENT if the display is from a trace log data set for a still-active trace or a complete trace that has not yet been purged. You can view the active trace data still in the buffers by selecting the CURRENT button.

Note: In most cases, HISTORY will show more data than CURRENT. But just after a log switch, there can be more data in the online display buffers than is available on this new data set. However, the History Traces application also allows you to go back to read the previous log data set.

Target Specification

The target for the History Traces application can be any target that is controlled by this BBI-SS PAS. If the target of the trace in the selected data set is different from the current application target, LTRAC, STRAC, and DTRAC show the historical target in the HIST TGT field. The standard TGT field continues to show the target value from the History Traces panel.

Using the History Traces Application (Option 5)

The History Traces application displays a menu with one row per log data set in the directory. The initial display is in descending order of date and time but can be sorted by any column. For example, sort by TRACEID if you want to see all the data sets for one trace together. Sample displays are shown in Figure 9 and Figure 10:

```
BMC SOFTWARE ----------------- HISTORY TRACES  --------------  PERFORMANCE MGMT
COMMAND ===>                                                 TGT ===> DB2D
                                        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.TRACEDIR
ENTRIES USED: 1,209  FREE:   201
                                                              SCROLL RIGHT >>>
LC  DATE------TIME  TRACEID  TITLE                    USERID   TGT   STAT  ACTV
    01/03/29 22:00  LEOTST01 LEOS DETAIL TRACE        CIR4     DB2D  INV
    01/03/11 21:00  DET1     WORKLOAD DETAIL 1        CIR2     DB2D  USED  READ
    01/03/01 00:00  THRDHIST THREAD HISTORY           BABUSERS DB2D  UPDAT WRIT
    *************************** END OF DATA ***********************************
```

Figure 9. History Traces Application (Before Scrolling Right)

Scroll right to see additional data:

```
BMC SOFTWARE ------------------ HISTORY TRACES --------------  PERFORMANCE MGMT
COMMAND ===>                                                 TGT ===> DB2D
                                        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.TRACEDIR
ENTRIES USED: 1,209  FREE:   201
                                                               SCROLL LEFT <<<
LC  DATE------TIME  TYPE  DSN                                        VOLSER
    01/03/29 22:00  DET   CIR4.LL1X.DB2D.LEOTST01.MAR28.T2200.V01    ARCHIV
    01/03/11 21:00  DET   CIR21234.WORKLOAD.DETAIL1X.TEST1234.V02     BAB001
    01/03/01 00:00  SUM   CIR2.BL1X.DB1D.THRDHIST.MAR01.T0000.V01     BAB303
    *************************** END OF DATA ***********************************
```

Figure 10. History Traces Application (After Scrolling Right)

Field Definitions

DIRECTORY — The DSN of the directory.

ENTRIES USED — The number of entries in the directory that have been used.

FREE — The number of entries in the directory that have not been used. This field is highlighted if fewer than 10 entries remain.

Note: You can then use the batch maintenance jobs described in the next section to clean up or expand the directory.

DATE----TIME — Date and time that logging was activated on this data set.

TRACEID — The trace identifier that was specified when the trace was requested.

TITLE — The title specified when the trace was requested.

USERID — The ID of the user that requested the trace.

TGT — The target that was specified when the trace was requested. This is the DB2 subsystem or CICS region (if MAINVIEW for CICS is installed) on which the trace was made.

STAT — Status of log data set contents. Possible values can be as follows:

- INV — The data set is not a valid trace log data set. The exact reason for the failure is contained in a message written to the BBI journal.
- NOCAT — The data set is not cataloged.
- RESET — The data set contains valid trace data and has been flagged for reuse.
- UPDAT — The data set is being updated.
- USED — The data set contains valid trace data and was successfully closed.
- INCMP — The data set was not successfully closed (or archived) after an update; for example, in a system failure.
- EMPTY — The data set contains no trace records.

ACTV — Status of log data set activity. Possible values can be as follows:

- READ — Data set is being read by one or more users, but is not being written to.
- WRITE — The data set is being written to by an active trace.
- R/W — The data set is being written to by an active trace and also is being read by one or more users.
- ALLC — The data set is allocated to a currently active trace.
- blank — The data set is not currently in use.

TYPE	The type of trace: SUM (summary), DET (detail), or I/O (detail with I/O events).
DSN	The log data set name.
VOLSER	The volume serial name of the first volume used for this data set.

Primary Commands

The following primary commands can be typed on the COMMAND line:

- SORT by any column. Any of the column data in the display can be sorted by typing the first two characters of the column heading with the SORT command:

DA	Sorts the list by the date and time that logging was activated on this data set.
	Note: DATE-TIME is treated as one field so you cannot sort by time without also sorting by date.
TR	Sorts the list by trace identifier.
TI	Sorts the list by title.
US	Sorts the list by user ID.
TG	Sorts the list by target.
ST	Sorts the list by status of log data set contents.
AC	Sorts the list by status of log data set activity.
TY	Sorts the list by type of trace.
DS	Sorts the list by log data set name.
VO	Sorts the list by volume serial name.

- LOCATE an entry by the current sort column (similar to ISPF LOCATE member command).
- NEW presents a data entry screen to manually add other trace data sets to the directory. For example, you can move them from one system to another. Type the DSN only; the application automatically verifies each log DSN entered.

Figure 11 shows a sample data entry panel that has been updated with four data sets to be added to the directory. The RESPONSE field shows the results of the processing after ENTER is pressed.

```
BMC SOFTWARE ----------- CREATE TRACE DSN ENTRIES -----------  PERFORMANCE MGMT
COMMAND ===>                                                  TGT ===> DB2D
                                                              TIME --  09:36:34

 DIRECTORY: CIR4.LL1X.TRACEDIR

 SPECIFY TRACE DATASET NAME(S), ONE NAME PER ROW.
 THE HEADER RECORD IN EACH DATA SET WILL BE READ TO CREATE THE DIRECTORY ENTRY.

        LOG DATA SET NAME                              RESPONSE
        ---------------------------------------------  -----------------------
    1.  CIR4.LL1X.DB2D.LEOTST01.MAR19.T2200.V02        OK
    2.  CIR4XXXX.NODE2XXX.NODE3XXX.NODE4XXX.V03        WARN - NOT TLDS
    3.  CIR4.DUPLICATE.ENTRY.V01                       ERROR - DUPLICATE
    4.  CIR4.TEST.DO.NOT.EXIST.V01                     WARN - DOES NOT EXIST
    5.
    6.
    7.
    8.
    9.
    10.

Press ENTER to confirm; END to cancel.
```

Figure 11. Panel for New Primary Command

- STOP to close/de-allocate the directory; all commands except START are disabled; no new logging request can be initiated. Currently active trace logging is allowed to continue.

  Note: PMACC=# authorization is required to use this command.

- START causes the directory to be allocated and opened as old. If successful, functions are (re)enabled and new logging requests can be initiated. If not, the application remains in a quiesced state.

  Note: PMACC=# authorization is required to use this command.

- TYPE can be used to list only the traces of the specified type. The possible types that can be specified are shown in the TYPE column. For example, to list only the detail traces (including the I/O traces), type on the COMMAND line:

  ```
  TYPE DET
  ```

  Type TYPE to return to the list of all the traces.

Line Commands

Line commands apply to a specific entry within the directory.

Note: Certain line commands are restricted based on your authorization. PMACC=# is required for the following line commands if your user ID does not match the user ID of the trace:

- Delete
- Free
- Reset
- Archive

The possible commands are

- **S (SELECT)** loads the selected log data set and displays the LTRAC screen. From there, you can access all other trace displays.
- **W (SHOW)** displays the selected log data set information. The directory entry is updated if necessary.
- **P (PRINT)** generates the print JCL you can submit to print trace reports from the log data set.

  Note: This command is available only with a terminal session under ISPF.

  The print job accepts many options to tailor the output to your needs. Only one report can be specified. The following panel is displayed to allow specification of these options. See Chapter 5, “Printing a Trace” on page 51 for an explanation of these options and for instructions on how to create a batch job to produce multiple reports.

```
BMC SOFTWARE ------------ 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  ==>            Time  ==>
 To   date  ==>            Time  ==>

   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 12. Batch Trace Print Panel

The following optional panel to provide/update the job statement is shown if UPDATE JOB is set to Y:

```
BMC SOFTWARE -------- BATCH TRACE PRINT JOB STATEMENT -------  PERFORMANCE MGMT
COMMAND ===>


     JOB STATEMENT:

===> //CIR2TPRT  JOB (3861),,CLASS=K,MSGCLASS=R
===> //*
===> //*
===> //*

    SYSOUT CLASS ===> R
```

Figure 13. Batch Trace Print Job Statement

The date and time selection fields are primed with the start and end date-time of the trace. All other fields are primed with the default value when first used. These fields are saved in your profile for later redisplay.

When you press ENTER, the data is validated (cursory validation only) and redisplayed.

The selections made are saved in your profile and used to initialize the fields the next time this panel is requested.

When you exit from this panel, the options specified are inserted into a pattern job stream using the ISPF file tailoring facility. The skeleton JCL can be found in member DZJPTRAC in the BBPROF data set. This member can be tailored to an individual user (UBBPROF data set) or site (SBBPROF data set). The result of this customization is shown in edit mode, allowing you to make any further changes needed.

Note: If you did not use AutoCustomization, the &DLIB variable in this member must be specified.

You can then save this JCL, submit it to the local reader, or CANcel the edit session.

Note: To execute the job, you must enter SUBMIT while in edit mode. This is not done automatically for you.

- **D (DELETE)** deletes the entry from the trace directory. If the entry is for a data set that is not cataloged, no confirmation is required. If the data set is cataloged, a confirmation screen similar to the sample in Figure 14 is displayed:

```
BMC SOFTWARE --------- TRACE DIRECTORY PURGE CONFIRMATION ---- PERFORMANCE MGMT
COMMAND ===>                                                  TGT ===> DB2D
                                                              TIME --  09:36:34

 LOG DATA SET NAME:   CIR4.LL1X.TEST1.V02
 TARGET:              DB2D
 USERID:              CIR4
 TRACEID:             BIGELAP
 VOLUME:              ARCHIV
 START DATE-TIME:     01/03/22 22:00
 STATUS:              INV

 Do you wish to also delete the log data set? ===> N  (Y/N)


 Press ENTER to confirm; END to cancel
```

Figure 14. Trace Directory Purge Confirmation Panel

- **E (RESET)** flags the log data set for reuse.
- **V (VERIFY)** matches the directory against the log data set and updates the directory with the log data set fields. If the data set is not cataloged, the status is changed to NOCAT. If there is any error reading the header information, the status is changed to INV (invalid).
- **N (NEW)** acts like NEW primary command, except the data entry screen is primed with the selected log DSN.
- **A (ARCHIVE)** starts the archive started task (STC) specified in the trace request (TRARCSTC=) for the specified log data set. If TRARCSTC has not been specified, the request is ignored.
- **F (FREE)** closes and deallocates a log data set that is not in WRITE status. This can be used to deallocate a log data set when another user is viewing it.

Batch Maintenance Jobs

The trace logging facility includes six sample jobs provided in BBSAMP The sample jobs to manage the trace directory are described in the *MAINVIEW for DB2 Customization Guide*. The four jobs to manage individual TLDSs are defined here.

Trace Log Data Set Jobs

Use the following members to

- Define a trace log data set
- Archive a trace log data set
- Restore an archived trace log data set
- Print a trace log data set

Defining a Trace Log Data Set

Use BBSAMP member JXT011 to define a trace log data set as an LDS using IDCAMS. This pre-allocates a TLDS that can then be referenced when starting the trace. Also, this formats the TLDS, reducing trace startup time when large log data sets are required. You will need to use the job if you do not have authority for dynamic TLDS allocation by the BBI-SS PAS. (See page 30.)

Archiving a Trace Log Data Set

Use BBSAMP member DZTBARC to set up a started task for automatic archiving of a TLDS. It is executed when a data set is full, when logging is switched to another data set, or when the trace completes. This procedure has two steps:

1. Unload the trace log data set to a GDG data set
2. PGM=JXTRES to mark the trace log data set as RESET in header

You can add a step to print the trace if desired.

You also can use this member as a pattern to create a job to archive or reset a log data set. It can be invoked from the History Traces panel if previously defined for a trace. Otherwise, it must be submitted manually.

Restoring an Archived Trace Log Data Set

Use BBSAMP member DZTBRLOD to restore an archived trace log data set. This job has two steps:

1. Define a VSAM linear data set
2. Load the archived data to the defined linear data set

You can add the linear data set to the online trace directory to view the contents online. Use the NEW command on the History Traces application (see "Using the History Traces Application (Option 5)" on page 41).

Printing a Trace Log Data Set

This job is defined in Chapter 5, “Printing a Trace” on page 51.

Chapter 5. Printing a Trace

This chapter describes the offline printing facility for trace data. This is provided by a batch utility job, DZTBTRAC. The primary input source is data collected by MAINVIEW for DB2 online traces and stored on log data sets. Accounting records and I/O trace records from SMF or GTF also can be printed. One or more reports can be printed in one pass of the data.

Note: A job to print a report from one trace log data set can be generated online from the History Traces application. (See P line command on page 45.)

Overview

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 also can 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.

Types of Reports

This facility provides many useful reports to assist you in analyzing the collected trace data. The types of reports available depend on whether just the DB2 accounting records (summary) or trace events (detail) are present.

Note: Report request syntax and the complete options are described in "Request Keywords" on page 60.

Summary Traces

These reports can be created from both summary and detail traces and SMF/GTF DB2 accounting records.

TSTAT=SUMMARY
: One page summary of all records traced

LTRAC=YES
: Accounting short trace with one line per accounting record traced

STRAC=SUMMARY, ACCTG, ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLLIO, DDF, PKG
: Accounting trace with selected sections per accounting record traced

TSUM=A
: Accounting summary by AUTHID

TSUM=C
: Accounting summary by connection name

TSUM=L
: Accounting summary by location

TSUM=P
Accounting summary by plan

TSUM=R
Accounting summary by correlation ID

TSUM=T
Accounting summary by time

Detail Traces

Reports that can be created from detail traces are

TSTAT=ALL
Summary of all records traced, plus a database summary (if scans were traced), a lock and I/O summary (if locks and/or I/O were traced), and a sort summary.

STRAC=ALL
Accounting long trace with one report per accounting record, plus an SQL summary, a database summary (if scans were traced), a lock and I/O summary (if locks and/or I/O were traced), and a sort summary.

STRAC=(SQL,SCAN)
SQL and database trace summary; SORTS and IOLOCK summaries and the SQL statement pop-ups (SQLPOPUP) can also be requested.

DTRAC=YES
Detail event trace (LEVEL=1 shows all SQL, LEVEL=2 or 3 shows additional events).

DTRAC=YES,POPUP=ALL
Long detail event trace with event pop-up details. POPUP=(event1, event2, event3) prints pop-ups only for the selected events.

POPUP=(event1, event2, event3)
Selected detail event pop-ups only.

Detail Traces with I/O Events

Reports that can be created from detail traces with I/O events or an SMF file that includes I/O events are

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" on page 69 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=K
: I/O analysis by plan and package (only from a detail trace including I/O and SQL events)

DBIO=L
: I/O analysis by location

DBIO=P
: I/O analysis by plan

DBIO=S
: I/O analysis by plan and package and SQL statement (only from a detail trace including I/O and SQL events)

DBIO=T
: I/O analysis by time interval

Primary Input - Trace Log Data Set(s) (TLDS)

The normal input to this process is the VSAM data set created by a trace request with logging. All TLDSs are tracked in a trace directory and can be accessed through the History Traces application. You can submit a print job for a TLDS from this application (see the description of the P command in "Request Keywords" on page 60).

You also can submit a batch print job directly, as described in this section.

- You can specify one or more TLDSs in a series of DD statements with the ddnames in the form TRACINxx. Only TLDS data sets can be specified in this manner.
- If the exact names of the trace data sets are not known, you can specify which trace is to be printed with the TRACEID keyword. In this case, you must also specify the directory data set with the TRACEDIR DD statement. It is recommended but not required that the TARGET, DATE, and TIME keywords also be used to further qualify which trace data sets should be selected. With this option, only one TRACEID can be specified for this batch job. It remains in effect across all report requests.

Printing from Archived TLDSs

An archived TLDS is produced by the archive utility DZTBVUNL. To print directly from this sequential data set without reloading it to a VSAM linear data set, specify the DSN on the ARCIN DD statement.

Secondary Input - SMF or GTF

This same batch utility 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. All the I/O analysis reports listed under "Detail Traces with I/O Events" on page 52 (except DBIO=K or DBIO=S) can be produced from an SMF or GTF file that includes the I/O trace records (IFCIDs 06-10). Other detail trace events are not extracted from SMF or GTF. Also, you cannot process SMF or GTF files in the same run with TLDSs.

Output Formats

Multiple reports can be produced from a single pass of the input data, each to a separate DD statement.

Single reports also can be made of a combination of LTRAC, STRAC, STRAC summaries, DTRAC, and DTRAC pop-ups. However, the STRAC summaries (for example, SQL and database) cannot be combined with DTRAC data. They can be requested in the same batch job, but as separate reports.

The output format is the same as the online displays; it is 80 characters wide with the first position reserved for the attribute character. For normal output, the data is printed in columns 2 through 80.

For wide output, the highest level requested is left aligned and lower levels are indented for ease of viewing. STRAC and DTRAC are indented 5 positions and the pop-up displays are indented 40 positions. For example, if DTRAC and pop-ups are the only reports requested, DTRAC is aligned on the left margin, and pop-ups are indented 40 positions. If LTRAC, STRAC, DTRAC, and pop-ups are all requested, LTRAC lines are aligned on the left margin, STRAC is indented 5, DTRAC 10, and pop-ups 50.

Controls

All control of the input and formatting is done through a series of keywords in the input job stream (or in a PDS member). This control includes format, input source, input selection/exclusion, and display selection.

JCL

The JCL to produce any of the reports is shown in Figure 15. This JCL is a single-step procedure in member DZTBTRAC in BBSAMP

```
//         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='HILVL.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
//RPTEVNT   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(DZJPFAIL)  LOCK/EDM FAILURES    (D-BASE)
//          DD DSN=&PFX.BBSAMP(DZJPDBIO)  I/O SUMMARY BY DB/TS (D-I/O)
//*
```

Figure 15. JCL to Print a Trace (DZTBTRAC)

PRINT

Specifies the name of the program as:

PGM=DZTBPRNT

and the region required to run the program.

This program supports a GMWK option in the PARM field, for example:

```
PGM=DZTBPRNT,PARM='GMWK=128K'
```

This option is used to increase the size of the summary work area needed to process trace data sets that contain a very large number of transactions. This option increases the summary work area for all services (LTRAC, STRAC, and so on).

The syntax rules for GMWK are similar to those for the MVS JCL REGION parameter. The size specified should be greater than 128K. Storage acquired with this option is below the 16M line and is limited by the largest private area available in the MVS system. When the GMWK option is used, the job's REGION size may need to be adjusted accordingly.

A general guideline for the size of GMWK is

```
GMWK = 1700 + (72 x number of unique transactions summarized)
```

If multiple report requests are executed in a single job step, one summary work area is acquired for each report. If the summary work area size is too large, any region size available can be exceeded very quickly.

STEPLIB DD

Defines the program library containing the DZTBPRNT load module.

SYSPRINT DD

Defines the data set for printing all input statements and program messages.

Note: The SYSPRINT DD statement must exist. If it is not found, a WTO is issued and the run is terminated.

SYSUDUMP DD

Defines the dump data set for problem determination.

STD1

Defines the ddname for the default report.

TRACINxx DD

Defines one or more trace log data sets (TLDSs) as input for trace data formatting. If TRACINxx is defined, TRACEDIR, SMFIN, ARCIN, and GTFIN are ignored. This DD must refer to a VSAM trace data set (a TLDS not archived to tape), written by the online trace.

Multiple TLDS data sets can be processed by specifying each in a DD statement with a ddname of TRACINxx, where xx is any valid set of characters. Each of the TLDSs is processed in the order it exists in the job stream.

TRACEDIR DD

Designates a trace directory data set.

If no TRACINxx DD statement is found, the TRACEDIR DD statement is processed.

The TRACEID keyword is required. All trace data sets produced with this trace ID are eligible for processing by reporting.

Note: It is encouraged but not required that TARGET, DATE, and TIME keywords further restrict which data sets are to be accessed; only those TLDSs which meet the target, date, and time criteria specified by one or more of the report requests are allocated.

SMFIN DD

Defines an unloaded SMF data set (or multiple concatenated data sets) containing DB2 accounting records as input for trace data formatting. TRACIN and TRACEDIR must be dummied or NULLFILE. If SMFIN is defined, GTFIN is ignored.

GTFIN DD

Defines a GTF data set (or multiple concatenated data sets) containing DB2 accounting records as input for trace data formatting. TRACIN, TRACEDIR, and SMFIN must be dummied or NULLFILE.

ARCIN DD

Defines an archived TLDS. TRACIN, TRACEDIR, SMFIN, and GTFIN must be dummied or NULLFILE.

Note: If none of the five input DD options are defined, trace information is obtained from the currently active SMF data set.

REPTDD1

Defines the ddname for a report request. The report request in the SYSIN stream defines the ddname of the output. There is no restriction on these ddnames other than uniqueness; they cannot duplicate other reserved ddnames.

Note: If the ddname specified for a report is missing, a diagnostic is written to SYSPRINT and the report is not produced.

SYSIN DD

Defines the input data set containing the report definitions.

Note: All input data is entered through the SYSIN DD statement.

If the SYSIN DD statement is missing, DUMMYed, or empty, the default report selection is used. (See "Default Entry" on page 71.)

Individual control statements specify how to read the input data set for each of the requested reports and how to format the output.

Output Format

All reports have common headers in the following format:

```
BMC SOFTWARE                                        MAIN TITLE OF REPORT                                        PAGE:       1
REPORT XXXXXXXX                                       SECONDARY TITLE                                           DATE: 17MAR01
                                                                                                                TIME:15:31:44

   DSNAME:    SYS5.PRODSYS.DB2D.BLANK.OCT03.T0501.V01
   VOLSER:    BAB033

   TITLE:     ACCOUNTING TRACE             TYPE:        DETAIL
   START:     19MAR01 05:01:00               GROUP SQL: N
   STOP:      16MAR01 23:58:55               SCANS:     Y
   STATUS:    USED                           I/O:       Y
   DATA LOST: NO                             LOCKS:     Y
   REQUESTOR: CIR8                           DDF:       Y
   TRACEID:                                  DDFVTAM:   Y
   TARGET:    DB2D

   DB2PLAN:
   DB2AUTH:
   DB2CONN:
   DB2CORR:
   DB2LOC:

EXCEPTION FILTERS:
 ELAP    =>          GETPAGE  =>          MAXLOCK  =>        SQLDDL   =>
 CPU     =>          PGUPD    =>          LOCKTBL  =>        SQLDYN   =>
 ABORT   =>          READIO   =>          INCRBIND =>        SQLCTL   =>
 UIDCOM  =>          LOCKSUSP =>          SQLSEL   =>        SQLFETCH =>
 GETRIO  =>          LOCKESCL =>          SQLUID   =>        SQLTOT   =>
 RLF     =>          TIMEOUT  =>          RIDFAIL  =>        CLAIMDR  =>
 PWAITIO =>          PWAITLK  =>          PWAITPF  =>        PWAITOT  =>
 PFREQS  =>          PFREADS  =>          HPFAILS  =>        PRLLGRP  =>
 PRLLRED =>          PRLLFALB =>                             =>

...FIRST DATA LINE (NEW PAGE IF NEWPAGE ^= NO)
 .
 .
 .
```

Figure 16. Report Header

This header is printed as each input data set is processed. If reports other than summary reports are requested, the headings are imbedded within the body of the reports. If only summary reports (TSTAT and TSUM) are requested, all heading blocks are shown before the summary reports.

The date and time shown at the top right reflect the time the report is produced. The date and time of the data can be seen both within the reports and from the START and STOP time of trace data recording on this data set (as shown in lines 6 and 7 of this header).

Lines 3 and 4 identify the input data sets, while the remaining lines show the options specified when the trace was made (see ATRAC for descriptions).

For either narrow or wide reports, the heading lines fill the entire width of the report. The values on the right are right-justified with the right margin. The two headings are centered in the report.

Within the body of the report, the values displayed are identical to the screens from the corresponding online trace display, such as LTRAC or STRAC. See Chapter 9, “Application Trace Displays” on page 109 for descriptions of these online trace displays. If LTRAC and/or DTRAC is requested, the column headings are shown at the start of each data stream and at the top of each page as secondary headings.

Controls for Reporting

The control statements to request each of the sets of reports are entered in the input stream under ddname SYSIN. They also can be members of a PDS. This section describes the structure of the request and the keywords used to make the request.

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. For example:

```
GLOBAL TITLE1='DAILY REPORT'
```

causes the title, DAILY REPORT, to be printed on the first line of each page of each report. You do not need to specify this keyword again for each individual report requested.

All keywords can be specified as GLOBAL except REPORTID and DDNAME. GLOBAL can be repeated anywhere in the input stream to reset any combination of keywords. Each keyword specified with GLOBAL remains in effect until overridden in another GLOBAL specification, A GLOBAL keyword can also be overridden for a single request by specifying it for that one REPORT.

Structure of a Report Request

Each report request is started with REPORT, followed by a series of keywords connected by commas, which define the options requested for the report. The report request must include the REPORTID keyword; this is the only required keyword. All other keywords have defaults which are assumed if the keyword is not given. The first blank in the series terminates the processing for that report. All of the following text is considered to be comments.

If a value such as a title has embedded blanks, enclose it in quotation marks.

A statement is continued to the next line by ending the line with a comma-blank combination. The remainder of the line is considered to be a comment; processing continues with the first non-blank character on the next line. Continuations can be done only where a comma would normally occur. Keywords and values cannot be continued.

A line starting with an asterisk in the first column is a comment line. It can be embedded within a series of continuations.

An example of two report requests is shown next. The keywords used in these requests are described in the next section.

Sample Report Requests

```
GLOBAL  TIME=1300-1500
*
*       BIND-TEXT and EXPLAIN FROM TSO
*
REPORT  REPORTID=DAILY1,LTRAC=YES,NEWPAGE=TRAN,
        POPUP=(BIND-TEXT,EXPLAIN,OPEN),
        CONNECT=TSO,
        TITLE1='DETAILS OF BIND AND EXPLAIN',
        TITLE2='WITH STATS FROM OPEN',
        DDN=OUTPUT1,WIDTH=WIDE       COMMENTS ABOUT THE REPORT
*
*       FULL EXAMINATION OF CERTAIN SPECIAL TRANSACTIONS
*
REPORT  REPORTID=SPECIALS,LTRAC=YES,STRAC=SUMMARY,DTRAC=YES,POPUP=
ALL,
        NEWPAGE=(TRAN,FIRSTEVENT),LEVEL=3,
        PLAN=(PAY+++++,ACCT1+++),CONNECT=IMSP,
        AUTHID=(USR1,USR5)
```

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.

Request Keywords

The requests shown above are a preview of some of the keywords used to request a report. This section describes a complete list of the keywords, their possible values, their defaults, and limitations on use.

If a keyword is entered multiple times, a diagnostic message is issued and the first value entered is used. If mutually exclusive keywords are entered, the relative priority of the keywords is defined within the keywords, and the higher priority is used to resolve the conflict.

For each keyword, the full spelling is shown. Some keywords can be abbreviated for ease of entry.

The select keywords are ANDed to select entries for print. For those keywords that accept a list of values, the values are ORed. The select keywords are defined in Table 4 on page 62.

If no reports are selected, the default report set is LTRAC=YES, TSTAT=SUMMARY. The report selection keywords are defined in Table 5 on page 65.

The groups of keywords are described in Table 3 on page 61.

Identification Options

The following keywords identify a report by its name, title, and source of input:

Table 3. Identification Keywords

Keyword/Operand	Description
REPORTID={name}	**Required**. Each report must have a unique identification provided by REPORTID. This is the only required keyword. If the DDN keyword is not specified, REPORTID is also used as the ddname of the output data set. The value can be 1–8 characters, with any special characters allowed.
DDN|DDNAME=[ddname]	**Optional**. Defines the ddname that receives the report. Any 1–8 character name that satisfies the rules for ddname is allowed, other than the reserved names TRACINxx, TRACEDIR, SYSIN, and SYSPRINT. If not specified, the default is taken from the REPORTID keyword.
TRACEID=[id|BLANK]	**Optional**. Defines the ID of the trace that is to be matched in a search through the trace directory. All TLDS data sets that meet this requirement, plus any TARGET, TIME, and DATE requirements, are dynamically allocated and processed. To define a trace ID of blank, specify: `TRACEID=BLANK` **Note:** The DD statement TRACEDIR must be present before this keyword is processed. If the DD statement is not present, this keyword is ignored.
USERID=[id]	**Optional**. Defines the ID of the user who started the trace. All TLDS data sets that meet this requirement, plus any TARGET, TIME, and DATE requirements, are dynamically allocated and processed. **Note:** The DD statement TRACEDIR must be present before this keyword is processed. If the DD statement is not present, this keyword is ignored.
TARGET|TGT=[tgtname]	**Optional**. Defines the target system for a trace that is to be matched in a search through the trace directory. All TLDS data sets that meet this requirement, the TRACEID qualifier, plus any TIME and DATE requirements, are dynamically allocated and processed. If no TARGET is specified, all targets for the specified trace are selected for processing. TARGET is required for DBIOx reports. **Note:** The DD statement TRACEDIR must be present before this keyword is processed. If the DD statement is not present, this keyword is ignored.

Table 3. Identification Keywords (Continued)

Keyword/Operand	Description
TITLE1=[c.....c]	**Optional**. Title value is printed centered on the first line of each page of the report. The value can be up to 60 characters. If this keyword is not included, the title is left blank on the report.
TITLE2=[c.....c]	**Optional**. Secondary title is printed centered on the second line of each page of the report. The value can be up to 60 characters. If this keyword is not included, the secondary title is taken from the title specified when the trace was created. If the input is from SMF or GTF, the default value is blanks.

Selection Options

The following keywords define selection criteria used to select only certain transactions for display and summarization:

Table 4. Selection Keywords

Keyword/Operand	Description
PLAN=[plan name]	**Optional**. Plan name can be used to qualify which records are displayed and/or summarized. Up to 50 plans can be entered, enclosed in parentheses, and separated by commas. Either simple names or qualified names can be entered in any combination. Default is to select all records.
DB2PKG=[package name]	**Optional**. Package name can be used to qualify which records are displayed and/or summarized. Up to 50 packages can be entered, enclosed in parentheses, and separated by commas. Either simple names or qualified names can be entered in any combination. Default is to select all records.
AUTHID=[id]	**Optional**. Authorization ID can be used to qualify which records are displayed and/or summarized. Up to 50 authorization IDs can be entered, enclosed in parentheses, and separated by commas. Either simple names or qualified names can be entered in any combination. Default is to select all records.
CONNECT=[connection name]	**Optional**. Connection name can be used to qualify which records are displayed and/or summarized. Up to 50 connection names can be entered, enclosed in parentheses, and separated by commas. Either simple names or qualified names can be entered in any combination. Default is to select all records.

Table 4. Selection Keywords (Continued)

Keyword/Operand	Description
CORR=[correlation id]	**Optional**. Correlation ID can be used to qualify which records are displayed and/or summarized. Up to 50 correlation IDs can be entered, enclosed in parentheses, and separated by commas. Either simple names or qualified names can be entered in any combination. Default is to select all records.
LOC=[location id]	**Optional**. Location ID can be used to qualify which records are displayed and/or summarized. Up to 10 location IDs can be entered, enclosed in parentheses, and separated by commas. Either simple names or qualified names can be entered in any combination. Default is to select all records.

DATE=[ddmmmyyyy|
ddmmmyyyy-ddmmmyyyy]
	Optional. Date value is used to select a date range or starting date from the input. The input can be a single date in ddmmmyy format, which defines a starting day, or as a range in the form ddmmmyyyy-ddmmmyyyy to define starting and ending dates to process.
	
	The asterisk is used to specify current date. To select all entries from yesterday, specify:
	
	`DATE=*_1`
	
	Note: The underscore is used instead of a dash, since the dash is already used to define a range of dates. Using *-1 would result in interpreting the date as a range from current date to day 1 (which is invalid as a date).
	
	To process only one day, specify:
	
	`17MAR2001-17MAR2001`
	
	Default is to select all records.

Table 4. Selection Keywords (Continued)

Keyword/Operand	Description

TIME=[hhmm|hhmm-hhmm]	**Optional**. Time value is used to select a starting time or a time range for the input.
	
	To select records starting at noon, specify:
	
	`TIME=1200`
	
	To select records from first shift operations only, specify:
	
	`TIME=0800-1559`
	
	To select records up to 4pm, specify:
	
	`TIME=-1600`
	
	To select records for a 24-hour period on the same day, specify:
	
	`TIME=0000-2359`
	
	or
	
	`TIME=0001-2400`
	
	Default is to select all records.

DBTS=[(dbname,tsname)]	**Optional**. Database name and table space name can be used to qualify which I/O event records are displayed and summarized in DBIOx displays. Only one can be entered. Either simple names or qualified names can be entered. Default is to select all I/O event records.
BP=[BPnn]	**Optional**. Buffer pool ID can be used to qualify which I/O event records are displayed and summarized in the DBIOx displays. Any number of buffer pool IDs can be entered, enclosed in parentheses, and separated by commas. Default is to select all I/O event records.

Report Options

The following keywords define which reports are to be produced:

Table 5. Report Keywords

Keyword/Operand	Description
LTRAC=[YES|NO]	**Optional**. Is used to request the LTRAC one-line entry for each transaction. If used in conjunction with STRAC, DTRAC, and/or POPUP, each transaction starts with the LTRAC line followed by the other displays as requested. Possible values are YES and NO. Default is LTRAC=NO.
STRAC=[SUMMARY|YES|ACCTG| ALL|NO|ENV|ELAPSED| SQLCOUNTS|BPOOL|LOCKS| PRLLIO|DDF|PKG|PKGPOPUP| SPAS|SQL|SQLPOPUP|SCAN| IOLOCK|SORTS]	**Optional**. Is used to request the STRAC transaction summary display. STRAC=NO suppresses the STRAC displays. STRAC=SUMMARY or YES requests only the first section of the STRAC display. STRAC=ACCTG requests all summary trace sections. These requests can be combined with any of the DTRAC and pop-up requests. STRAC=ALL requests all summary accounting sections plus detail event summaries as well as the transaction summary. **Note:** STRAC=ALL cannot be combined with any DTRAC or pop-up request. (Recall that multiple reports can be produced from one pass of the data. STRAC=ALL can be requested in a second report.) Each of the individual segments of STRAC also can be requested separately. The possible values are ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLLIO, DDF, PKG, SQL, SCAN, IOLOCK, and SORTS. The last four requests cannot be combined with DTRAC or pop-up requests. STRAC=SQL prints the SQL summary section. STRAC= SQLPOPUP prints the pop-ups of the individual SQL statement statistics. STRAC=PKG prints the Package / DBRM Overview section. STRAC=PKGPOPUP prints the pop-ups of the individual packages or DBRMs. Default is STRAC=NO.
SORTSQL=[PGM|ST|CO|AE|%E| AC|%C|SR|PI|PD|PW|PR|COP| %EP|%CP|SRP|PIP|PDP|PWP|PRP]	**Optional**. Controls the order of display within the STRAC SQL Summary section. It applies only to this summary and to the corresponding SQL statement pop-ups. The values and defaults are the same as defined for the STRAC SQL Summary display (see "Sort" on page 244).

Table 5. Report Keywords (Continued)

Keyword/Operand	Description
SORTSCAN=[DA|IS|IP|IR|DS|DP|DR|WS|WP|WR]	**Optional**. Controls the order of display within the STRAC Database Summary section. It applies only to this summary. The values and defaults are the same as defined for the STRAC Database Summary display (see "Sort" on page 249).
SORTIOLOCK=[DP|LM|LS|LT|SR|SW|SE|SA|AR|AP]	**Optional**. Controls the order of display within the STRAC Database Lock and I/O Summary section. It applies only to this summary. The values and defaults are the same as defined for the STRAC Database Lock and I/O Summary display (see "Sort" on page 251).
DTRAC=[YES|NO]	**Optional**. Requests printing a one-line entry for each detail trace event that meets the LEVEL criteria. For ease of associating the detail data with a transaction, it is recommended that either LTRAC or STRAC also be requested, but this is not required. Possible values are YES and NO. Default is DTRAC=NO.
LEVEL=[1|2|3]	**Optional**. Controls which detail entries are to be displayed in the DTRAC and pop-up displays. The possible values are 1, 2, or 3. The meaning of the levels is the same as in the DTRAC display. **Note:** When POPUP is specified without DTRAC, LEVEL only takes effect with POPUP=ALL. Default value is 2.
POPUP=[ALL|YES|NO|SQL|(event1, event2,,, eventn)]	**Optional**. Specifies which detail pop-ups are to be displayed. It can be specified with DTRAC or separately. POPUP=NO requests that no pop-ups be shown. POPUP=SQL requests that all pop-ups for SQL events be shown. POPUP=ALL or YES requests all possible pop-ups be shown. (The message, THERE IS NO POPUP FOR THIS EVENT, is not shown.) Any combination of pop-ups can be requested by listing their names as shown in the EVENT column of the DTRAC display. Multiple entries are enclosed in parentheses and separated by commas. Up to 50 events can be individually requested. Default is POPUP=NO.

Table 5. Report Keywords (Continued)

Keyword/Operand	Description
TSTAT=[SUMMARY|ALL|YES|NO]	**Optional**. Requests the summary Trace Statistics display. TSTAT=SUMMARY requests only the first 21 lines of the display. TSTAT=ALL or YES requests all lines of the TSTAT display. TSTAT=NO suppresses the summary display. If requested in association with any of the LTRAC, STRAC, DTRAC, or pop-up displays, this display follows all of these other displays. Default is TSTAT=NO.
TSUM=[A|C|L|P|R|T]	**Optional**. Requests one of the summary displays. The acceptable values for TSUM= are A, C, L, P, R, and T. If Y is entered, it is interpreted as TSUM=P. If requested in association with any of the LTRAC, STRAC, DTRAC, or pop-up displays, this display follows all of these other displays. Default is to not produce any TSUM report.
INTERVAL|INTVL|I=[nM|nH]	**Optional**. Is used in the TSUMT display to define the size of the summary interval. The value entered can be any 1–3 digit numeric value followed by M(inutes) or H(ours). If the suffix is omitted, hours is assumed. Default value is 1H.
SORTTSUM=[AU|CO|PL| IS|ISD| EC|AE|AC|A#|AG|TE|TC|T#|TG]	**Optional**. Controls the order of display within the TSUMx reports by sorting on the specified column. It applies only to those reports. The values and defaults are the same as defined for each of the interactive TSUMx displays. For example, SORTTSUM=TC sorts by TOTAL CPU.
GRAPH=[AVG|TOT|NO]	**Optional**. Requests the graphic presentation of any of the TSUMx displays, showing the Entry Count (number of threads) followed by either the averages (GRAPH=AVG) or totals (GRAPH=TOT). GRAPH=NO requests a numerical display of both average and total values. Default is GRAPH=NO.

Table 5. Report Keywords (Continued)

Keyword/Operand	Description
DBIO=[X|A|C|F|K|L|P|S|T]	**Optional**. Requests one of the I/O Analysis displays. X selects the DBIO display by DB/TS. The other values select the display with the select code of DBIOx. This determines the summary key(s) of the report: • X - Database and table space • A - AUTHID • C - Connect • F - Buffer pool • K - Plan and package • L - Location • P - Plan • S - Plan and package and SQL statement • T - Time interval See "Special Report Summarization Options for DBIO" on page 69.
IOINTVL|IOI=[nM|nH]	**Optional**. Is used in the DBIOT display to define the size of the summary interval. The value entered can be any 1–3 digit numeric value followed by M(inutes) or H(ours). If the suffix is omitted, hours is assumed. Default is 10 minutes.
IOSORT=[IC|I%|MI|AI]	**Optional**. Controls the order of display within the DBIOx reports by sorting on the specified column. It applies only to those reports. These values allow sorting on any of the data columns, as defined for the interactive displays. Default is to sort by the summary key(s).
IODATA=[SYNC|ASYNC|<u>TOTAL</u>]	**Optional**. Determines the type of I/O events selected for the DBIOx reports: SYNC Synchronous I/O events ASYNC Asynchronous I/O events TOTAL All I/O events Default is TOTAL.

Special Report Summarization Options for DBIO

The following keywords define the special report summarization options for the DBIO report:

Table 6. Special Summarization Keywords for DBIO

Keyword/Operand	Description
DBIO=[XA|XC|XF|XK|XL|XP|XT]	**Optional**. DBIO keyword with a two-character operand of Xx requests a special version of the DBIO display that provides reports by DB/TS plus another summary key. This allows you to determine where the activity against the table spaces is originating. **Note:** This determination is done online with repetitive summarization and selection steps. The following summary key combinations are supported: • XA - Database, table space, AUTHID • XC - Database, table space, connect • XF - Database, table space, buffer pool • XK - Database, table space, plan, package • XL - Database, table space, location • XP - Database, table space, plan • XT - Database, table space, time It cannot be requested together with any other trace displays. WIDTH=WIDE is required. Default is to not produce any DBIO report.
DBIO=[AX|CX|FX|KX|LX|PX|TX]	**Optional**. DBIO keyword with a two-character operand of xX requests a special version of the DBIO display that provides reports a summary key plus DB/TS. This allows you to determine which table spaces are being accessed from that source. **Note:** This determination is done online with repetitive summarization and selection steps. The following summary key combinations are supported: • AX - AUTHID, database, table space • CX - Connect, database, table space • FX - Buffer pool, database, table space • KX - Plan, package, database, table space • LX - Location, database, table space • PX - Plan, database, table space • TX - Time, database, table space It cannot be requested together with any other trace displays. WIDTH=WIDE is required. Default is to not produce any DBIO report.

Formatting Options

The following keywords define the formatting options for the selected report:

Table 7. Formatting Keywords

Keyword/Operand	Description
NEWPAGE=[TRAN|EVENT|FIRSTEVENT|NO]	**Optional**. Is used to control page breaks. The possible values are TRAN — Page break for each new transaction. EVENT — Page break for each new event or detail trace entry. FIRSTEVENT — Page break for the first event of a transaction. This can be used to separate the DTRAC entries from an STRAC or LTRAC display. NO — No page breaks are to be done based on input data. These entries can be entered in combinations, such as NEWPAGE=(TRAN,FIRSTEVENT). Default is NEWPAGE=NO.
WIDTH=[N|NARROW|W|WIDE]	**Optional**. Defines whether the titles are aligned for 81 character lines (narrow) or 133 character lines (wide). To specify wide output, use WIDTH=W or WIDE; to specify 81 character headings, use WIDTH=N or NARROW. You can specify any LRECL and BLKSIZE you desire on the JCL. RECFM=FBA is specified internally and cannot be overridden. If you do not specify LRECL, it defaults to either 81 or 133. If you do not specify a BLKSIZE, it defaults to LRECL. Default is WIDTH=WIDE.
LINECNT=[nnn]	**Optional**. Defines number of lines per page. The minimum number of lines allowed per page is 8, to allow for pre-defined headings; the maximum number allowed is 999. If LINECNT=0, the headings are printed only on the first page; no further page breaks are taken. Default value is LINECNT=66.

Table 7. Formatting Keywords (Continued)

Keyword/Operand	Description
HEADING=[YES|NO]	**Optional**. If all headings including the first heading should not be printed, use HEADING=NO with LINECNT=0. HEADING=NO suppresses all headings. If HEADING=NO,LINECNT=nn is specified, a new page is requested every nn lines but no headings are printed.
MAXPAGES=[nnnnnn]	**Optional**. To prevent unexpectedly high volumes of output, MAXPAGES can be specified. This stops detail report production when the specified number of pages has been reached per report. The summary processing continues; no limit is placed on the number of pages written by the summaries (TSUMx and TSTAT). Any numeric value up to 999,999 can be specified. A value of zero allows any number of pages to be printed. Default value is MAXPAGES=0.

Default Entry

If no report request is entered (SYSIN is empty, DUMMYed, or missing), a default request is constructed as follows:

```
REPORTID=STD1,
         TITLE1='TRACE ENTRIES',
```

This request uses the defaults as described above. Since no report selection is stated, it uses a default selection of LTRAC=YES,TSTAT=SUMMARY.

Return Codes

The return set by a batch process is used by operations, scheduling routines, and others to determine the success of an execution. The return codes set by this function are as follows:

- 0 - All processing successfully completed.
- 4 - Warning messages issued. Defaults taken to bypass error condition. Probably all output delivered successfully.
- 8 - Error conditions have been detected. Some output was delivered successfully, but there is a strong possibility that not all requested output was delivered correctly.
- 12 - Serious error. No proper output delivered.
- 16 - Internal processing error. Processing terminated, probably through no fault of the user.

For example, if two reports are requested and one of them has errors in the report definition, the return code is 8. If only one is requested and it has errors in the report definition which cause no output to be produced, the return code is 12.

Examples of Output

Sample report statements are provided in BBSAMP as DZJPxxxx members. Additional options are also described in these members.

The reports you can produce with these sample members are

- DZJPSUM—Accounting Summary Report

  provides

  - Trace totals
  - Trace summary by plan, sorted by plan name

- DZJPWKLD—Workload Summary by Time

  for a thread summary by 15-minute intervals showing totals in a graphic format.

- DZJPACCL—Accounting Detail—Long

  for complete accounting data per thread.

- DZJPACCS—Accounting Detail—Short

  for basic accounting data per thread.

- DZJPSQL—SQL Statement Summary

  for basic accounting data per thread, followed by:

  - An SQL statement summary sorted by percent CPU used
  - The SQL statistics summary pop-ups per statement

- DZJPEVNT—Thread Detail Events

  for a one-line identification of each thread, followed by a chronological list of level 1 and 2 events (critical and SQL events).

- DZJPEXPL—SQL Text and EXPLAIN

  for a one-line identification of each thread, followed by detail event pop-ups showing:

  - BIND parameters
  - Full SQL statement text and EXPLAIN data for all threads with dynamic SQL or the execution of a BIND request

- DZJPFAIL—Lock and EDM Pool Failures

  for a one-line identification of each thread, followed by detail event pop-ups for any lock or EDM-full failures, including complete deadlock and timeout details.

- DZJPDBIO—I/O Summary by DB/TS

  for an I/O report summarized by database and table space.

Following are examples of the output for these report requests.

Accounting Summary Report

Use sample member DZJPSUM to produce the Accounting Summary Report shown in Figure 17. It shows trace totals, followed by a trace summary by plan.

```
BMC SOFTWARE                ACCOUNTING SUMMARY REPORT               PAGE:       2
REPORT:SUM                                                          DATE: 17MAR01
                                                                    TIME:12:16:56
- - - - - - - - - - SUMMARY STATISTICS - ALL TRACE ENTRIES- - - - - - - - - - -
-----TERMINATIONS ----------    ---------ACTIVITY-----------------------------
FIRST END..01MAR 14.37.35.21                TOTAL   AVERAGE   MAXIMUM   MINIMUM
LAST END...01MAR 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
-- IFI 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
----------------   ------- --------- -------- ------- 0 ...25...50...75..100%
ELAPSED TIME                                           |                      |
  IN DB2                 4  00:02:10 00:08:42   84.15 | ****************     |
  IN APPLICATION         4     25 s  00:01:38   15.84 | ***                  |
 --TOTALS--              4  00:02:35 00:10:20  100.00 | ******************** |
WAITS IN DB2 (LOCAL)                                   |                      |
  LOCK/LATCH           184    125 ms    23 s     3.70 | <                    |
  I/O WAIT             115     62 ms 7,120 ms    1.14 | <                    |
  OTHER READ         2,757    133 ms 00:06:07   59.15 | ***********          |
  OTHER WRITE            0      0 us     0 us    0.00 |                      |
  UNIT SWITCH           33    375 ms    12 s     1.99 | <                    |
  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                 20    210 ms 4,202 ms    0.67 | <                    |
  MSG. PROCESSING        0      0 us     0 us    0.00 |                      |
 --TOTAL WAITS--     3,109    133 ms 00:06:53   66.67 | *************        |
*NOT ACCOUNTED                       00:01:36   15.40 | ***                  |
...........

BMC SOFTWARE                ACCOUNTING SUMMARY REPORT               PAGE:       4
REPORT:SUM                                                          DATE: 17MAR01
                                                                    TIME:12:16:56
                            I/O SYSTEM TRACE
          ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
PLAN      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
DSN8CC0       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 17. Accounting Summary Report

Workload Summary by Time

Use sample member DZJPWKLD to produce the Workload Summary by Time Report shown in Figure 18.

This report shows a thread summary by one-minute intervals, but any time span can be specified (default = 15 minutes). Use `GRAPH=AVG` to easily detect application exceptions. Use `GRAPH=TOT` for total workload analysis.

```
BMC SOFTWARE                WORKLOAD SUMMARY BY TIME              PAGE:       2
REPORT:WKLD                                                       DATE: 17MAR01
                                                                  TIME:13:40:03
                           I/O SYSTEM TRACE
INTERVAL     #THREADS    TOT ELAPSED      TOT CPU    TOT #STMTS    TOT GETPGS
START               6      00:01:11      6,300 ms           273         1,601
--------  |----.-----|  |----.-----|  |----.-----|  |----.-----|  |----.-----|
18:22:00   **           **********    <            <            <
18:23:00
18:24:00
18:25:00   **           *********     <            <            <
18:26:00   *******      *             *****        **********   **********
18:27:00   **********   <             **           ***          ******
18:28:00   *****        ***           **********   *            **
************************** END OF SUMMARY ENTRIES ***************************
```

Figure 18. Workload Summary by Time

Accounting Detail—Long

Use sample member DZJPACCL to produce the Accounting Detail—Long Report shown in Figure 19 and Figure 20 on page 76. It shows complete accounting data per thread.

```
BMC SOFTWARE                ACCOUNTING DETAIL - LONG              PAGE:       2
REPORT:ACCL                                                       DATE: 17MAR01
                                                                  TIME:12:17:09
STOP.....27JUN 18.26.15.25 PLAN................RXDB2 TYPE.............ALLIED
START....27JUN 18.26.09.17 AUTHID...............CIR1 CONNECT......DB2CALL/CAF
ELAPSED..........6,086 ms ORIG PRIM AUTH.......CIR1 CORR ID.....CIR1
TERM........NORMAL/DEALLOC COMMITS.................0 ROLLBACKS.............1
-------------------------------------------------------------------------------
RUNTIME ANALYSIS   IN DB2     IN APPL.      TOTAL      %IN DB2(=)     TOTAL(*)
----------------   --------   --------   --------     0 ...25...50...75..100%
ELAPSED TIME       2,520 ms   3,565 ms   6,086 ms     | ========************ |
CPU TIME             663 ms   1,986 ms   2,649 ms     | ==******             |
DB2 WAIT TIME      1,051 ms                            | ===                  |
- - - - - - ACTIVITY - - - - - -      - - - - - KEY INDICATORS  - - - - - - -
TOTAL SQL.....................245     I/O RSP: SYNC= 6,754 us, ASYNC=    10 ms
GETPAGES...................1,160
SYNC I/O (PRLL= 0) ...........21
PREFETCH......................82
UPDATES/COMMIT...............0.0
BFR HIT RATIOS:...VP= 98%,HP=  0%

- - - - - - - - - - - - - ENVIRONMENTAL INDICATORS - - - - - - - - - - - - - -
LUWID..........USBOOL01.LUDB2F.A97E4E8174E40001
RLF TABLE ID....NOT ACTIVE

- - - - - - - - ELAPSED TIME ANALYSIS (ACCTG CLASSES 2,3 ONLY) - - - - - - - -

CATEGORY           #EVENTS AVG/EVENT  ELAPSED  %TOTAL
----------------   ------- --------- -------- ------- 0 ...25...50...75..100
ELAPSED TIME                                           |                      |
  IN DB2                             2,520 ms   41.41 | ********             |
  IN APPLICATION                     3,565 ms   58.58 | ***********          |
 --TOTALS--                          6,086 ms  100.00 | ******************** |
WAITS IN DB2                                           |                      |
  LOCK/LATCH            38    250 us 9,517 us    0.15 | <                    |
  I/O WAIT              42  6,754 us   284 ms    4.66 | <                    |
  OTHER READ            72     10 ms   752 ms   12.35 | **                   |
  OTHER WRITE            0      0 us     0 us    0.00 |                      |
  UNIT SWITCH            2  2,889 us 5,779 us    0.09 | <                    |
  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 |                      |
 --TOTAL WAITS--       154  6,824 us 1,051 ms   17.27 | ***                  |
*NOT ACCOUNTED                            93 us    0.00 |                      |

- - - - - - - - - - - - SQL STATEMENT EXECUTION COUNTS - - - - - - - - - - - -
SELECT...............0    LOCK TABLE...............0    CREATE............0
INSERT...............0    GRANT/REVOKE.............0    DROP..............0
UPDATE...............0                                 ALTER.............0
DELETE...............0    SET CURR.SQLID...........0
                           SET HOST VAR. ...........0
PREPARE..............13   SET CURR.DEGREE..........0
DESCRIBE.............0
                           CONNECT TYPE 1...........0    COMMENT ON.........0
OPEN.................13   CONNECT TYPE 2...........0    LABEL ON...........0
FETCH................206  SET CONNECTION...........0
CLOSE................13   RELEASE..................0

** DML..............245   ** DCL...................0    ** DDL............0
```

Figure 19. Accounting Detail—Long—Part I

```
BMC SOFTWARE                ACCOUNTING DETAIL - LONG              PAGE:       3
REPORT:ACCL                                                       DATE: 17MAR01
                                                                  TIME:12:17:09

- - - - - - - - - - - - - - BUFFER POOL ACTIVITY - - - - - - - - - - - - - - -
ACTIVITY                TOTAL       BP0
------------------   --------  --------
GETPAGES..........      1,160     1,160
SYNC READS........         21        21
GETPAGES/READIO...       55.2      55.2
COND. GP FAILURES.          0         0

SEQ. PREFETCH REQS         82        82
LIST PREFETCH REQS          0         0
DYNAMIC PREFETCHES          0         0
ASYNC PAGES READ..          0       421
PAGES/PREFETCH REQ       14.1      14.1

PAGE UPDATES......          0         0
IMMEDIATE WRITES..          0         0

HP SYNC READS.....          0         0
HP SYNC READ FAIL.          0         0
HP ASYNC PAGES READ         0         0
HP WRITES.........          0         0
HP WRITE FAILURES.          0         0

- - - - - - - - - - - - - - - - LOCK ACTIVITY  - - - - - - - - - - - - - - - -
TIMEOUT....................0          MAX PAGE LOCKS.................21
DEADLOCK...................0
                                       LOCK REQUEST..................639
SUSPEND-LOCK...............0          UNLOCK REQUEST................350
SUSPEND-LATCH.............13          QUERY REQUEST...................0
SUSPEND-OTHER..............0          CHANGE REQUEST................13
                                       OTHER REQUEST...................0
ESCALATION(SHR)............0
ESCALATION(EXCL)...........0

DRAIN REQUEST..............0          DRAIN FAILURE...................0
CLAIM REQUEST..............9          CLAIM FAILURE...................0

- - - - - - - - PACKAGE / DBRM OVERVIEW (ACCTG CLASSES 7,8 ONLY) - - - - - - -

PACKAGE/                                          %TOTAL
DBRM      NO. SQL  CPU TIME  WAIT TIM  ELAPSED   ELAP.
-------- --------  --------  --------  -------- ------ 0 ...25...50...75..100
RXSEL1M         5     40 ms    181 ms    610 ms  10.02 | **                  |
RXSEL2M       104    208 ms    742 ms  1,351 ms  22.20 | ****                |
RXSEL3M        24     51 ms     37 ms    100 ms   1.64 | <                   |
RXSEL4M         8     31 ms    386 us     35 ms   0.56 | <                   |
RXSEL5M         9     32 ms  2,549 us     37 ms   0.60 | <                   |
RXSEL6M        10     35 ms    257 us     43 ms   0.70 | <                   |
RXSEL7M        11     34 ms     26 ms     39 ms   0.63 | <                   |
RXSEL8M        12     35 ms  1,595 us     70 ms   1.14 | <                   |
RXSEL9M        13     36 ms  2,585 us     47 ms   0.76 | <                   |
RXSELAM        14     38 ms    318 us     42 ms   0.68 | <                   |
RXSELBM        15     40 ms  2,468 us     45 ms   0.74 | <                   |
RXSELCM        16     40 ms  1,884 us     45 ms   0.74 | <                   |
RXSELDM        17     43 ms  7,401 us     56 ms   0.92 | <                   |
```

Figure 20. Accounting Detail—Long—Part II

Accounting Detail—Short

Use sample member DZJPACCS to produce the Accounting Detail—Short Report shown in Figure 21. It shows basic accounting data per thread.

```
BMC SOFTWARE               ACCOUNTING DETAIL - SHORT               PAGE:       2
REPORT:ACCS                                                        DATE: 17MAR01
                                                                   TIME:12:17:16
STOP.....27JUN 18.26.15.25 PLAN................RXDB2 TYPE.............ALLIED
START....27JUN 18.26.09.17 AUTHID..............CIR1 CONNECT......DB2CALL/CAF
ELAPSED..........6,086 ms ORIG PRIM AUTH.......CIR1 CORR ID.....CIR1
TERM........NORMAL/DEALLOC COMMITS.................0 ROLLBACKS.............1
-------------------------------------------------------------------------------
RUNTIME ANALYSIS   IN DB2     IN APPL.      TOTAL      %IN DB2(=)     TOTAL(*)
----------------   --------   --------   --------     0 ...25...50...75..100%
ELAPSED TIME       2,520 ms   3,565 ms   6,086 ms     | ========************ |
CPU TIME             663 ms   1,986 ms   2,649 ms     | ==******             |
DB2 WAIT TIME      1,051 ms                            | ===                  |
- - - - - - ACTIVITY - - - - - -      - - - - - KEY INDICATORS  - - - - - - -
TOTAL SQL....................245     I/O RSP: SYNC= 6,754 us, ASYNC=    10 ms
GETPAGES...................1,160
SYNC I/O (PRLL= 0) ...........21
PREFETCH......................82
UPDATES/COMMIT...............0.0
BFR HIT RATIOS:...VP= 98%,HP=  0%
```

Figure 21. Accounting Detail—Short

SQL Statement Summary

Use sample member DZJPSQL to produce the SQL Statement Summary Report shown in Figure 22. It shows basic accounting data per thread, followed by:

- An SQL statement summary sorted by percent CPU used
- The SQL statistics summary pop-ups per statement

```
BMC SOFTWARE                                          SQL STATEMENT SUMMARY                                    PAGE:       2
REPORT:SQL                                                                                                     DATE: 17MAR01
                                                                                                               TIME:12:17:23

STOP.....27JUN 18.22.37.08 PLAN...............DSN8CC0 TYPE..............ALLIED
START....27JUN 18.21.26.40 AUTHID..............T305 CONNECT....CICS41B1/CICS
ELAPSED...........00:01:11 ORIG PRIM AUTH........T305 CORR ID.....GT00D8CS
TERM........NORMAL/DEALLOC COMMITS.................0 ROLLBACKS.............1
-------------------------------------------------------------------------------
RUNTIME ANALYSIS   IN DB2     IN APPL.      TOTAL      %IN DB2(=)     TOTAL(*)
----------------   --------   --------   --------     0 ...25...50...75..100%
ELAPSED TIME       00:01:11      98 ms   00:01:11     | ===================* |
CPU TIME              17 ms      35 ms      52 ms     | <                   |
DB2 WAIT TIME      00:01:10                           | ===================  |
- - - - - - ACTIVITY - - - - - -      - - - - - KEY INDICATORS  - - - - - - -
TOTAL SQL......................12     TIMEOUT
GETPAGES.......................7     LOCK SUSPENSIONS =      1
SYNC I/O (PRLL= 0) ............0
PREFETCH.......................0
UPDATES/COMMIT................0.0
BFR HIT RATIOS:...VP=100%,HP=  0%

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

  STMT                  AVG.     %      AVG.     %    SORT  -- PAGES SCANNED --
  TYPE   STMT  COUNT   ELAPSED  ELAP    CPU     CPU   RECS  INDX DATA WORK  REF
------- ----- ------  -------- -----  -------- -----  ----  ---- ---- ---- ----
FETCH     962      5  2,136 us  46.6    410 us  36.6     0     1    1    0    0
PGM:DSN8CC1        5            46.6            36.6     0     1    1    0    0

SELECT    814      1  9,664 us  42.2  1,363 us  24.3     0     1    0    0    0
PGM:DSN8CC0        1            42.2            24.3     0     1    0    0    0

SELECT    931      1  1,074 us   4.7    697 us  12.4     0     1    0    0    0
SELECT   1021      1    689 us   3.0    690 us  12.3     0     1    1    0    0
SELECT    670      1    540 us   2.4    541 us   9.7     0     1    0    0    0
CLOSE     973      1    131 us   0.6    129 us   2.3     0     0    0    0    0
OPEN      956      1    130 us   0.6    128 us   2.3     0     0    0    0    0
INSERT    802      1      0 us   0.0      0 us   0.0     0     0    0    0    0
PGM:DSN8CC1        6            11.2            39.0     0     3    1    0    0

** TOTALS ***     12                                     0     5    2    0    0
          ---------------------------------------------------------------------------
          |STATEMENT: 962 FETCH    CURSOR                    NUMBER OF EXECUTIONS:      5 |
          |PLAN:      DSN8CC0                ELAPSED:  AVERAGE 2,136 us   TOTAL    11 ms |
          |PROGRAM:   DSN8CC1                CPU:      AVERAGE   410 us   TOTAL 2,050 us |
          |                                                                             |
          |                                          --------- AVERAGES ---------       |
          |                                             INDEX  SEQ-DATA                 |
          |                                          --------  --------                 |
          |        ROWS PROCESSED ALL TYPES              12.0       0.8                 |
          |        ROWS PROCESSED CORRECT TYPE           12.0       0.8                 |
          |        ROWS QUALIFIED BY DM (STAGE 1)         0.8       0.0                 |
          |        ROWS QUALIFIED BY RDS (STAGE 2)        0.8       0.0                 |
          |        ROWS INSERTED                          0.0       0.0                 |
          |        ROWS UPDATED                           0.0       0.0                 |
          |        ROWS DELETED                           0.0       0.0                 |
          |        PAGES SCANNED                          0.2       0.2                 |
          |       REFERENTIAL INTEGRITY PROCESSING:                                     |
          |        PAGES SCANNED                          0.0       0.0                 |
          |        ROWS DELETED/SET NULL                  0.0       0.0                 |
          ---------------------------------------------------------------------------
```

Figure 22. SQL Statement Summary

Thread Detail Events

Use sample member DZJPEVNT to produce the Thread Detail Events Report shown in Figure 23. It shows a one-line identification of each thread, followed by a chronological list of detail events. Level 3 events such as scans and I/O can also be included.

```
BMC SOFTWARE                                          THREAD DETAIL EVENTS                                    PAGE:       2
REPORT:EVNT                                                                                                   DATE: 17MAR01
                                                                                                              TIME:12:17:28
  19MAR01                 I/O SYSTEM TRACE
 END TIME     PLAN    AUTHID  CONNECT   ELAPSED   CPU    # STMTS GETPAGE REASON
----------- -------- -------- -------- -------- -------- ------- ------- ------
18:22:37    DSN8CC0  T305     CICS41B1 00:01:11    52 ms      12       7  OK
         EVENT        AT     ELAPSED     CPU   DETAIL
     -------------- -------- -------- -------- -------------------------------------
     CREATE-THD        0.000    38 ms 1,602 us
      PLAN-ALLOC       0.037                     ISOLATION=CS ACQ=USE   REL=COMMIT
     PKG-ALLOC         0.041                    *DSN8CC0  ISO=CS ACQ=USE   REL=COMIT
     SELECT    814     0.041 9,664 us 1,363 us *RC( 100)              X     PS(   1)
     PKG-ALLOC         0.079                    *DSN8CC0  ISO=CS ACQ=USE   REL=COMIT
     SELECT    670A    0.079   540 us   541 us *RC( 100)              X     PS(   1)
     SELECT    931A    0.082 1,074 us   697 us *RC( 100)              X     PS(   1)
     OPEN      956A    0.084   130 us   128 us *RC(   0) C=VO2
     FETCH  (5)962A    0.093    11 ms 2,050 us *RC( 100) C=VO2        D/X   PS(   2)
     CLOSE     973A    0.116   131 us   129 us *RC(   0) C=VO2
     SELECT   1021A    0.124   689 us   690 us *RC(   0)              D/X   PS(   2)
     INSERT    802A    0.128             0 us *RC(   0)
      LOCK-TMO-DET    70.375                    *DB=dsn8d71P OB=DSN8S71C NUMBER=1
      LOCK-TMO        70.376                    *DB=dsn8d71P OB=DSN8S71C TYP=ACE
      ABORT           70.701    13 ms   842 us
        LOCK-SUMMARY  70.713                    *MAXPG(0) ESCL(0) TS(  1)
        COMMIT-LSN    70.713                    *LOCK-AVOID=N PAGESETS=0
      TERM-THD        70.714             0 us

BMC SOFTWARE                                          THREAD DETAIL EVENTS                                    PAGE:       3
REPORT:EVNT                                                                                                   DATE: 17MAR01
                                                                                                              TIME:12:17:28
  19MAR01                 I/O SYSTEM TRACE
 END TIME     PLAN    AUTHID  CONNECT   ELAPSED   CPU    # STMTS GETPAGE REASON
----------- -------- -------- -------- -------- -------- ------- ------- ------
18:25:51    DSN8CC0  T305     CICS41B1 00:01:03    47 ms      12       7  OK
         EVENT        AT     ELAPSED     CPU   DETAIL
     -------------- -------- -------- -------- -------------------------------------
     CREATE-THD        0.000 4,634 us 1,469 us
      PLAN-ALLOC       0.004                     ISOLATION=CS ACQ=USE   REL=COMMIT
     PKG-ALLOC         0.008                    *DSN8CC0  ISO=CS ACQ=USE   REL=COMIT
     SELECT    814     0.009    10 ms 1,450 us *RC( 100)              X     PS(   1)
     PKG-ALLOC         0.025                    *DSN8CC0  ISO=CS ACQ=USE   REL=COMIT
     SELECT    670A    0.026   558 us   559 us *RC( 100)              X     PS(   1)
     SELECT    931A    0.032 1,331 us   839 us *RC( 100)              X     PS(   1)
     OPEN      956A    0.035   137 us   134 us *RC(   0) C=VO2
     FETCH  (5)962A    0.037 2,857 us 1,960 us *RC( 100) C=VO2        D/X   PS(   2)
     CLOSE     973A    0.048   119 us   119 us *RC(   0) C=VO2
     SELECT   1021A    0.050   712 us   712 us *RC(   0)              D/X   PS(   2)
     INSERT    802A    0.053             0 us *RC(   0)
      LOCK-TMO-DET    62.631                    *DB=dsn8d71P OB=DSN8S71C NUMBER=1
      LOCK-TMO        62.655                    *DB=dsn8d71P OB=DSN8S71C TYP=ACE
      ABORT           62.811    13 ms   814 us
        LOCK-SUMMARY  62.823                    *MAXPG(0) ESCL(0) TS(  1)
        COMMIT-LSN    62.823                    *LOCK-AVOID=N PAGESETS=0
      TERM-THD        62.824             0 us
```

Figure 23. Thread Detail Events

SQL Text and EXPLAIN

Use sample member DZJPEXPL to produce the SQL Text and EXPLAIN Report shown in Figure 24. It shows a one-line identification of each thread, followed by detail event pop-ups showing:

- BIND parameters
- Full SQL statement text and EXPLAIN data for all threads with dynamic SQL or the execution of a BIND request

```
BMC SOFTWARE                                        SQL TEXT AND EXPLAIN DATA                                PAGE:       2
REPORT:REPORTA                                                                                               DATE: 17MAR01
                                                                                                             TIME:10:59:09
  19MAR01                 DB2 APPLICATION TRACE
 END TIME     PLAN    AUTHID  CONNECT   ELAPSED   CPU    # STMTS GETPAGE REASON
----------- -------- -------- -------- -------- -------- ------- ------- ------
14:50:15    DSNTEP41 BOLSMR4  BATCH      844 ms   227 ms      13      57  OK
         EVENT        AT     ELAPSED     CPU   DETAIL
     -------------- -------- -------- -------- -------------------------------------
      BIND-TEXT        0.012                  *TYPE=DYNAMIC  TEXT=EXPLAIN PLAN SET+
                                          ---------------------------------------------------------------------------
                                          |EXPLAIN PLAN SET QUERYNO=03 FOR SELECT X.ACTNO, X.PROJNO, Y.JOB, Y.EDLEV       |
                                          |EL FROM DSN8710.EMPPROJACT X, DSN8710.EMP Y WHERE X.EMPNO = Y.EMPNO AND        |
                                          |X.EMPTIME > 0.5 AND (Y.JOB = 'DESIGNER' OR Y.EDLEVEL >= 12) ORDER BY X.A       |
                                          |X.EMPTIME > 0.5 AND (Y.JOB = 'DES
                                          |CTNO, X.PROJNO                                                                 |
                                          ---------------------------------------------------------------------------
      EXPLAIN          0.087                  *PLAN=DSNTEP41 COST(54.9)
                                          ---------------------------------------------------------------------------
                                          |       QUERY NUMBER  3              TIMESTAMP      2001-03-19  14:50:1455       |
                                          |       GROUP MEMBER  DB2G           COLLECTION ID  DSNTEP2                     |
                                          |                                                                               |
                                          | BLK  SEQ DESCRIPTION                                                          |
                                          |---- ---- ----------------------------------------------------------------------|
                                          |   1    1 METHOD: FIRST TABLE                                                  |
                                          |          ACCESS: SEQUENTIAL SCAN          PREFETCH: SEQ                        |
                                          |          TSLOCK: IS                                                            |
                                          |          TABLE:  EMP                CORR: Y                  CREATOR: DSN8710 |
                                          |          INDEX:                                              CREATOR:         |
                                          |          SORT:   NONE                                                          |
                                          |---- ---- ----------------------------------------------------------------------|
                                          |   1    2 METHOD: HYBRID JOIN                                                  |
                                          |          ACCESS: INDEX SCAN               PREFETCH: LIST                       |
                                          |          TSLOCK: IS                                                            |
                                          |          TABLE:  EMPPROJACT         CORR: X                  CREATOR: DSN8710 |
                                          |          INDEX:  XEMPPROJACT2       MATCHING 1 INDEX KEY     CREATOR: DSN8710 |
                                          |          SORT:   NONE                                                          |
                                          |---- ---- ----------------------------------------------------------------------|
                                          |   1    3 METHOD: ADDT'L SORT                                                  |
                                          |          ACCESS:                          PREFETCH: NONE                       |
                                          |          TSLOCK:                                                               |
                                          |          TABLE:                                              CREATOR:         |
                                          |          INDEX:                                              CREATOR:         |
                                          |          SORT:   COMPOSITE TABLE    ORDER BY                                   |
                                          ---------------------------------------------------------------------------
      BIND-TEXT        0.106                  *TYPE=DYNAMIC  TEXT=SELECT * FROM PL+
                                          ---------------------------------------------------------------------------
                                          |SELECT * FROM PLAN_TABLE                                                       |
                                          ---------------------------------------------------------------------------
      EXPLAIN          0.193                  *PLAN=DSNTEP41 COST(4,521.4)
                                          ---------------------------------------------------------------------------
                                          |       QUERY NUMBER  1,008          TIMESTAMP      2001-03-19  14:50:1464       |
                                          |       GROUP MEMBER  DB2G           COLLECTION ID  DSNTEP2                     |
                                          |                                                                               |
                                          | BLK  SEQ DESCRIPTION                                                          |
                                          |---- ---- ----------------------------------------------------------------------|
                                          |   1    1 METHOD: FIRST TABLE                                                  |
                                          |          ACCESS: SEQUENTIAL SCAN          PREFETCH: SEQ                        |
                                          |          TSLOCK: IS                                                            |
                                          |          TABLE:  PLAN_TABLE                                  CREATOR: BOLSMR4 |
                                          |          INDEX:                                              CREATOR:         |
                                          |          SORT:   NONE                                                          |
                                          ---------------------------------------------------------------------------
```

Figure 24. SQL Text and EXPLAIN

Lock and EDM Pool Failures

Use sample member DZJPFAIL to produce the Lock and EDM Pool Failures Report shown in Figure 25. It shows a one-line identification of each thread, followed by detail event pop-ups for any lock or EDM-full failures, including complete deadlock and timeout details.

```
BMC SOFTWARE                                        LOCK AND EDM POOL FAILURES                                PAGE:       2
REPORT:REPORTA                                                                                                DATE: 17MAR01
                                                                                                              TIME:10:59:43
  19MAR01                 DB2 APPLICATION TRACE
 END TIME     PLAN    AUTHID  CONNECT   ELAPSED   CPU    # STMTS GETPAGE REASON
----------- -------- -------- -------- -------- -------- ------- ------- ------
11:16:15    RXDB2    BOLBPL3  DB2CALL     16 s    208 ms       5      53  OK
         EVENT        AT     ELAPSED     CPU   DETAIL
     -------------- -------- -------- -------- -------------------------------------
     TIMEOUT          15.360                    *DB=DSN8D71P OB=DSN8S71C NUMBER=1
                                          ------------------------------------------------------------------------------
                                          |             LOCK TYPE:      DATA PAGE                                       |
                                          |             RESOURCE NAME:  DB=DSN8D71P OB=DSN8S71C                          |
                                          |             RESOURCE #:     X'00001200'                                     |
                                          |             HASH VALUE:     X'01041300'                                     |
                                          |             FUNCTION:       LOCK                                            |
                                          |             STATE:          X                                               |
                                          |             DURATION:       MANUAL                                          |
                                          |             ZPARM INTERVAL:  10                                             |
                                          |             INTERVAL COUNT:  1                                              |
                                          |             REQUESTING WU:  01790057-06D25738                               |
                                          | ---------------------LOCK HOLDERS / PRIORITY WAITERS -------------------------|
                                          | H/W    PLAN     CORRID/LUWID CONNECT  DURATION     STATE MEMBER   OWNER WU    |
                                          | ------ -------- ------------ -------- ------------ ----- -------- ------------|
                                          | HOLDER RXDB2    BOLBPL2      DB2CALL  COMMIT       X     DB2G     01790054-   |
                                          |             USBOOL01LUDB2GABC2CBA6EB8F0001                          06D25458    |
                                          ------------------------------------------------------------------------------
```

Figure 25. Lock and EDM Pool Failures

I/O Summary by DB/TS

Use sample member DZJPDBIO to produce the I/O Summary by DB/TS Report shown in Figure 26. It shows an I/O report summarized by database and table space. The sample also shows how to produce reports with a third summary key, such as database, table space, plan or plan, database, table space.

```
BMC SOFTWARE                I/O ANALYSIS BY DB/TS                PAGE:       1
REPORT:DBIO                                                      DATE: 17MAR2001
                                                                 TIME:12:17:48
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  DSNSCT02             1   1.4     28     28 | *******              |
DSNDB01  DSNSPT01             1   1.4     17     17 | ****                 |
DSNDB01  SCT02                2   2.8     38     31 | *******              |
DSNDB01  SPT01                4   5.6    351     95 | ******************** |
DSNDB06  DSNAPH01            29  40.8   1028     82 | ******************** |
DSNDB06  DSNDSX01             2   2.8     27     27 | ******               |
DSNDB06  DSNDTX01             1   1.4     40     40 | **********           |
DSNDB06  SYSDBASE             1   1.4     26     26 | ******               |
```

Figure 26. I/O Summary by DB/TS

Chapter 6. Solving Problems

This chapter analyzes several problem symptoms you may encounter when using a trace. Each symptom is listed, followed by an analysis of the problem and a suggested solution.

Symptoms:

- A trace request shows a status of QIS instead of ACTV.
- Message DZ3101I TRACE QUIESCED is written to the Journal Log.
- A message IM47nnx with the reason for quiescing is written.

Analysis/Solutions:

There are several reasons a trace could be quiesced:

- The target DB2 is no longer active:
  - All current trace requests (and monitors) have a status of QIS.

  Solution:

  You can check if the target DB2 is inactive by looking at the DB2 messages in the Journal Log (Option L, or PF5), or simply by requesting one of the online displays such as DB2ST. If the RST (restart) option for the trace is HOT, the trace is automatically reactivated when DB2 restarts.
- TRACE LIMIT EXCEEDED

  A detail trace was requested, but the maximum of four for that DB2 are already active.

  Solution:

  One of the four detail traces must be stopped (not necessarily purged) before another can be started.
- DB2 TRACE FAILURE

  A detail trace was requested, but the start trace request to DB2 failed.

  Solution:

  Check the Journal Log for additional DB2 messages to determine and correct the error.
- TRACE LOGGING QUIESCED

  Logging was requested, but logging is not active because the Trace Directory is not available.

  Solution:

  Check the Journal Log at the time the BBI-SS PAS was started. Analyze the additional messages about the failure to allocate the Trace Directory. Correct the problem and restart the BBI-SS PAS to activate logging.

- TLDS ALLOCATION FAILED

  Logging was requested, but the log data set allocation failed.

  Solution:

  Check the Journal Log at the time the trace was started for additional messages about the allocation failure. Correct the error and resubmit the trace request.

- GETMAIN RETURN AREA FAILURE

  A GETMAIN for buffer storage failed because the BBI-SS PAS region size was exceeded.

  Solution:

  Reduce the size of STORAGE, TRSIZE, or TRBUFF to reduce the total amount of storage requested, or stop or purge other Current Traces to free up their storage allocations. Remember that a very large buffer GETMAIN (for example, STORAGE=8000K) may fail because of space fragmentation.

Symptom:

- A trace request fails with the short message:

  ```
  BUFFERS EXCEED LIMIT
  ```

Analysis:

The specifications for buffer sizes exceeded the allowed limit (TRLIM) for a single trace.

Solution:

Reduce the size of STORAGE, TRSIZE or TRBUFF to reduce the total amount of storage requested. The value for TRLIM is specified in BBPARM member DMRBEX00.

Symptom:

- Extended CSA usage is too high, and several buffers contain DB2 trace data (identifier RMID=26).

Analysis:

When MAINVIEW for DB2 issues a DB2 Start Trace request, DB2 GETMAINs storage in ECSA and allocates it to the BBI-SS PAS. The storage is only freed when DB2 terminates. The storage is reused when that OPx destination is reused for another trace request.

Solution:

Ensure that you have not specified too high a value for IFIBUFF. The default value is 32K. A maximum of 8 buffers of IFIBUFF size may be held by DB2.

Symptoms:

- DB2 issues the following message:

  ```
  DSNW133I  DSNWVSRV  TRACE DATA LOST, OPx NOT ACCESSIBLE, RC=8
  ```

  followed later by:

  ```
  DSNW123I  DSNWVSRV  TRACE RECORDING HAS BEEN RESUMED ON OPx
  ```

- When viewing the trace displays (STRAC, DTRAC) some of the detail events are missing, and the following message may be displayed near the top of the screen:

  ```
  DZ3114W  SOME DETAIL DATA LOST - BUFFER INCOMPLETE
  ```

Analysis:

MAINVIEW for DB2 cannot transfer the trace data passed from the DB2 IFI quickly enough. This may be because of dispatching priority or the volume of events.

Solution:

1. The BBI-SS PAS must be at a higher dispatching priority than DB2. Check that this is true.
2. This trace produces so many events that the DB2 IFI buffer is filling up too quickly.
   - Remove some of the event types traced (especially LOCKS, DDFVTAM).

   Or:

   - Temporarily increase the size of the IFI buffer before starting this trace (proper authorization is required). This can be done with an undocumented parameter IFIBUFF in DMRBEX00 for the target DB2.
     - Define a new member DMRBEX01 in BBPARM with IFIBUFF=200K to 400K.
     - Issue the BBI control command RESET PARM DMRBEX01.
     - Start the trace.
     - Issue the BBI control command RESET PARM DMRBEX00 to return to the default 32K for other trace requests.

Symptom:

- When viewing a detail trace (STRAC or DTRAC displays), one of the following messages is seen:

```
DZ3114W  SOME DTL TRACE DATA LOST - NO BUFFER
DZ3114W  SOME DTL TRACE DATA LOST - BUFFER OVERFLOW
DZ3114W  SOME DTL TRACE DATA LOST - BUFFER INCOMPLETE
```

Analysis / Solutions:

When collecting detail trace data, an insufficient number or size of buffers was available to collect all of the event data being passed from DB2.

- NO BUFFER

  No buffer was available to collect the trace events for one or more threads. Only the accounting record is available for viewing for this thread (or threads).

  Solution:

  – Increase the number of buffers to three times the expected number of concurrent threads.

  Or:

  – Restrict the number of concurrent threads that could be traced with the trace selection criteria DB2AUTH and/or DB2PLAN.

- BUFFER OVERFLOW

  At least once while a thread was being traced, the current buffer filled up and no additional buffer was available to continue. Tracing continues when a buffer does become available, but detail events in the middle/end of that thread are missing.

  Solution:

  Increase TRBUFF or TRSIZE, or reduce the number of events to be traced. This problem is often caused by tracing LOCKS or DDFVTAM events, which can be overwhelming in number. If you need to trace these events without logging, restrict the number of concurrent threads to one with selection criteria, specify TRBUFF=3, and increase TRSIZE.

  If you specify logging, make sure you have three to five buffers per concurrent thread and keep the size as usual. Three buffers may not be enough to trace a single thread when running on a fast CPU and logging many events. In this case, the I/O to write a buffer and free it up for reuse may not complete quickly enough. If you are tracing multiple threads (for example, 5 possible threads * 3 = 15 buffers), then the buffer usage and I/Os occur at different times for each thread; this increases the chances of finding a free buffer and 3 probably will be sufficient.

  Note: See the suggestions for "Analyzing Lock Usage Without Tracing LOCK Events" on page 26.

- BUFFER INCOMPLETE

  Although sufficient buffer storage was available, at least some detail events are missing for one or more threads.

  – The trace was started or stopped in the middle of thread processing—the first or last events are missing. If stopped before thread completion, only a dummy accounting record without statistics is available.

    Solution:

    Start/stop the trace when no threads are processing, or ignore the incomplete threads.

  – The data from DB2 was arriving faster than it could be processed.

    Solution:

    Check the Journal Log for message DSNW133I to verify that this is the problem (dispatching priority or size of IFIBUFF) and refer to that symptom description.

  – When viewing a long-running application from Current Traces (the online STORAGE buffer), the first events for some threads are missing because only the last two buffers per thread can be saved online.

    Solution:

    View this trace from History Traces (the trace log data sets), where all buffers can be saved and combined to show the complete thread data.

  – When viewing a transaction that spans multiple trace log data sets, only the last log data set contains DB2 accounting statistics. Detail trace data (DTRAC, UTRAC, and detail trace summaries in STRAC and TSTAT) is incomplete in each log data set.

    However, accounting data in the final log is complete and represents the entire transaction. The detail trace data in each log can be added together for total results.

    Solution:

    When tracing applications that can span multiple buffers, allocating one large trace log data set is preferable to several small ones.

Symptom:

- No data is shown in an active trace for a long-running thread.

Analysis / Solution:

The data is being collected, but is usually only shown in LTRAC/STRAC when the thread completes and the DB2 accounting record is available. While the thread is active, you can view the events with the UTRAC display, which is like DTRAC, but it is accessible from DUSER for that thread.

You can set up trace logging for long-running threads, specifying one or more small trace logs.

Each time a trace log fills up, a dummy accounting record is written for any threads being traced that are still incomplete. These entries then show up on LTRAC (with a status of INCMP) and can be selected for viewing. The STRAC display obviously has none of the accounting data, but does have the detail summaries of events (SQL, IO/LOCK..). From STRAC, you can expand to the DTRAC entries.

If the trace is not stopped, the final detail events and the accounting record information appear in the active trace log when the thread completes, as before.

Specify LOG WRAP=YES or NO, depending on which part of the thread you want to keep. With YES, the earlier events can be overwritten.

Depending on the size of your STORAGE buffer (for online display), you can also view the data from the previous logs from the History Traces application.

If the trace is stopped before the thread completes, the dummy accounting record is not written and the last trace data set will not be usable. You can use the log switch command to force a trace data set switch before stopping the trace. The log switch command will write the dummy accounting record to the end of the current trace data set before switching.

Part 2. Application Trace

This part describes the application trace services.

Chapter 7. Introduction 93

Chapter 8. Application Trace Considerations 95
Data Collection 95
Display Level Options 97
Usage Examples 98
(Application Developer) Test Application Performance 98
(DBA) Review SQL Before Production Use 100
(DBA) Review Application Performance Periodically 101
(DBA) Investigate Problem Plans 102
(DBA) Analyze Index Usage and Database Access Paths 102
(DBA) Trace QMF and Ad Hoc SQL Usage 103
(DBA) Analysis of Short-Term Workload History 103
(DBA) Analyze I/O by Page Set 105
(DBA) Analyze Lock Contention 106
(System Programmer) Check New Production Application Performance 107
(System Programmer) Set Up Ongoing Exception Traces per Workload Type 108

Chapter 9. Application Trace Displays 109
Trace Display Summary 109
Navigating Between Trace Displays 111

Chapter 10. Viewing a List of Completed Threads 115
LTRAC—DB2 Trace Entries 115

Chapter 11. Viewing Summary Data for All Trace Entries 121
TSTAT—Trace Statistics 121
Summary Statistics (Base Section) 130
Accounting Sections 139
Elapsed Time Analysis 139
SQL Statement Execution Counts 144
Buffer Pool Activity 148
Lock Activity 151
Parallelism 154
Routines 157
DDF Summary 158
Detail Trace Summary Sections 161
Database Summary 161
Database Lock and I/O Summary 162
Sort Summary 164

Chapter 12. Viewing Workload History Summaries 165
TSUMx—Trace Summary Displays 165
Display Fields 165
Service Messages 167
Scrolling 167
Using EXPAND 167
Displaying Graphics 168
Sorting the Display 169
Selecting a Subset of Trace Entries 171
TSUMA—Trace Summary by Authorization ID (AUTHID) 172
TSUMC—Trace Summary by Connection Name (CONNECT) 174
TSUML—Trace Summary by Location Name (LOCATION) 176
TSUMP—Trace Summary by Plan Name (PLAN) 178
TSUMR—Trace Summary by Correlation ID (CORR-ID) 180
TSUMT—Trace Summary by Time (INTERVAL START) 182

Chapter 13. Viewing Summary Data for a Specific Thread 185
STRAC—Summary Trace Entry 185
Thread Identifiers and Status Summary (Base Section) 198
Accounting Sections 210
Environmental Indicators 210
Elapsed Time Analysis 211
SQL Statement Execution Counts 216
Buffer Pool Activity 220
Lock Activity 223
Parallelism 226
Routines 229
DDF Summary 233
Package/DBRM Overview 237
Package/DBRM Pop-up 239
Detail Trace Summary Sections 242
SQL Summary 242
SQL Statement Pop-up 246
Database Summary 249
Database Lock and I/O Summary 251
Sort Summary (Detail Trace Only) 253

Chapter 14. Viewing All Events for a Specific Transaction 255
DTRAC—Detail Trace Entry 255
Display Fields 258
Group SQL 258
Event Data 259
Event Pop-Up Displays 274
Allocate Cursor Pop-Up (IFCID 273) 275
Associate Locators Pop-Up (IFCID 272) 276
Authorization Failure Pop-Up (IFCID 140) 277
Authorization Translation Pop-Up (IFCID 169) 278
Bind Pop-Up (IFCID 108/109) 279
Bind Text Pop-Up (IFCID 063) 283
Claim Pop-Up (IFCID 211) 284
Claim Wait Pop-Up (IFCID 215/216) 286
COMMIT-LSN Pop-Up (IFCID 218) 287
Conversation Queued Pop-Up (IFCID 167) 288
Conversation Request Pop-Up (IFCID 159) 289

Deadlock Pop-Up (IFCID 172) 290
Distributed SQL Pop-Up (IFCID 168) 293
Drain Pop-Up (IFCID 212) 294
Drain Wait Pop-Up (IFCID 213/214) 296
EXPLAIN Pop-Up (IFCID 022) 298
Free Pop-Up (IFCID 110/111) 314
Hybrid Join Failure Pop-Up (IFCID 190) 315
Lock Request Pop-Up (IFCID 021) 316
Lock Summary Pop-Up (IFCID 020) 318
Lock Suspension Pop-Up (IFCID 044/045) 320
Lock Timeout/Deadlock Pop-Up (IFCID 054) 322
Page P-Lock Pop-Up (IFCID 259) 323
Package Allocation Pop-Up (IFCID 177) 325
Parallel Degree Pop-Up (IFCID 221) 327
Parallel Elapsed Pop-Up (IFCID 222) 330
Parallel Group Pop-Up (IFCID 231) 332
Plan Allocation Pop-Up (IFCID 112) 334
P-Lock Pop-Up (IFCID 251) 335
Read I/O Pop-Up (IFCID 006/007) 339
Requesting Agent Data Pop-Up (IFCID 183) 340
Requesting Site—DRDS Pop-Up (IFCID 157) 342
Requesting Site—DTM (IFCID 162) 343
Requesting Site—VTAM Pop-Up (IFCID 160) 344
RID List Pop-Up (IFCID 125) 346
Scan Pop-Up (IFCID 017/018) 348
Serving Site—DRDS Pop-Up (IFCID 158) 350
Serving Site—DTM Pop-Up (IFCID 163) 351
Serving Site—VTAM Pop-Up (IFCID 161) 352
Sort Pop-Up (IFCID 95/96) 354
SQL Statement Pop-Up 356
Stored Procedures Entry Pop-Up (IFCID 233) 359
Stored Procedures Exit Pop-Up (IFCID 233) 360
Synchronous Write Pop-Up (IFCID 008/009) 361
Table Check Constraint Pop-Up (IFCID 305) 362
Temporary Table Pop-Up (IFCID 311) 364
Timeout Pop-Up (IFCID 196) 366
Trigger Entry Pop-Up (IFCID 325) 369
Trigger Exit Pop-Up (IFCID 325) 371
User-Defined Function Pop-Up (IFCID 324) 373
Utility Pop-Up 375
VTAM Exit Pop-Up (IFCID 164) 377
VTAM Macro Pop-Up (IFCID 165) 378
Wait I/O Pop-Up (IFCID 127/128) 379
UTRAC—User Detail Trace 380
Display Fields 381

Chapter 15. Viewing I/O Activity 383
DBIOx—I/O Analysis (Trace). 383
Display Fields. 384
Description 385
Service Message(s). 385
Using EXPAND 385
Sorting the Display. 386
Selecting a Subset of I/O Events 386
DBIO—I/O Analysis by Database / Table Space (DB/TS) 388
DBIOA—I/O Analysis by Authorization ID (AUTHID) 390
DBIOF—I/O Analysis by Buffer Pool (BPOOL) 391
DBIOC—I/O Analysis by Connection Name (CONNECT) 392
DBIOL—I/O Analysis by Location (LOCATION) 393
DBIOP—I/O Analysis by Plan (PLAN) 394
DBIOK—I/O Analysis by Package / Program (PKG/PGM) 396
DBIOS—I/O Analysis by SQL Statement (SQL STMT) 398
DBIOT—I/O Analysis by Time (INTERVAL START) 400

Chapter 7. Introduction

Application trace comprises the data collection and data display services shown in Table 8. These services track the processing of specific transactions through DB2 and display the timestamps, resource usage, and other important workload statistics for each transaction (DB2 accounting record) traced. The services use selected Instrumentation Facility Component IDs (IFCID) obtained through the DB2 Instrumentation Facility Interface (IFI).

Table 8. Application Trace Services

Service	Title	Page	Area
ATRAC	DB2 APPLICATION TRACE	13	Data Collection
LTRAC	DB2 TRACE ENTRIES	115	Trace Displays
TSTAT	TRACE STATISTICS	121	
TSUMA	TRACE SUMMARY BY AUTHID	172	
TSUMC	TRACE SUMMARY BY CONNECT	174	
TSUML	TRACE SUMMARY BY LOCATION	176	
TSUMP	TRACE SUMMARY BY PLAN	178	
TSUMR	TRACE SUMMARY BY CORRID	180	
TSUMT	TRACE SUMMARY BY TIME	182	
DBIO	I/O ANALYSIS BY DB/TS	388	
DBIOA	I/O ANALYSIS BY AUTHID	390	
DBIOF	I/O ANALYSIS BY BPOOL	391	
DBIOC	I/O ANALYSIS BY CONNECT	392	
DBIOL	I/O ANALYSIS BY LOCATION	393	
DBIOP	I/O ANALYSIS BY PLAN	394	
DBIOK	I/O ANALYSIS BY PKG/PGM	396	
DBIOS	I/O ANALYSIS BY SQL STMT	398	
DBIOT	I/O ANALYSIS BY TIME	400	
STRAC	SUMMARY TRACE ENTRY	185	
DTRAC	DETAIL TRACE ENTRY	255	
UTRAC	USER DETAIL TRACE	380	

The ATRAC service (DB2 APPLICATION TRACE) is a timer-driven service that can be activated multiple times (with different selection criteria and trace types) to trace selectable workloads. The display services show the data collected by an ATRAC request.

Note: Data collection for a trace must be activated by the ATRAC service before the data can be displayed by the application trace display services. The appropriate DB2 traces are activated through the Instrumentation Facility Interface (IFI).

A trace entry is made in an internal buffer for each DB2 accounting record selected with the ATRAC service request as described in Chapter 2, “Collecting Trace Data” on page 7. These entries are sometimes referred to as transactions, since this terminology is used generally by online systems such as IMS and CICS.

Chapter 8. Application Trace Considerations

This chapter summarizes how to use the summary and detail application traces. It provides overhead considerations, itemizes the data available for display at each detail trace display level, and explains how this information can be used by data management personnel.

Part 1, “Using a Trace” on page 1 describes how trace data is collected and how to request trace data collection, display, and print.

Data Collection

The many trace data collection levels provide flexibility to collect just that information needed to perform a specific task without incurring overhead for data that is not needed. SUMMARY and DETAIL are relatively inexpensive. Each additional type of event, SQL|SCAN|IO|LOCK|DDF|DDFVTAM, adds more overhead but gives more information. One or more of these groups can be added to a DETAIL trace. Security profiles control the level of trace each user is allowed to specify.

Trace overhead is directly proportional to the number of events traced. Keeping a detail trace active for a longer time period to ensure capturing several occurrences is not expensive if only a few executions of the traced plan are expected. However, tracing a long-running batch job or many transactions can be very expensive if a detail trace is chosen with several additional types of events.

Following are overhead considerations for each trace level. Each type of event specified includes additional data to that collected by a summary and detail trace.

Trace Type	Data Collected
SUMMARY	Accounting level data Minimal overhead, since DB2 accounting traces are usually active.
DETAIL	Adds major events Relatively low overhead because of selective performance tracing. DETAIL includes the same accounting data as summary.
SQL	Adds SQL statements to DETAIL Most useful level for application tuning. This group is included with DETAIL by default.
SCAN	Adds scans to DETAIL More overhead because of the potentially large number of scans, but it gives information on index usage and data access paths otherwise not available.
IO	Adds I/O to DETAIL More overhead because of the additional events traced (more storage is used to hold the large number of events). However, this information is valuable to show the status of the table organization by evaluating GETPAGE to READ I/O ratios.
LOCK	Adds lock requests to DETAIL Potentially, the greatest amount of overhead because of the large number of additional events traced (more storage is used to hold these events). It should be used only if locking is a problem. **Note:** The GROUPSQL option can be used to combine identical SQL statements processed in sequence to reduce buffer storage requirements for long-running batch jobs. For example, this could combine many FETCH statements into one line with summary counts.
DDF	Adds all DDF events except VTAM-related DDF events to DETAIL More overhead because of additional events traced.
DDFVTAM	Adds VTAM-related DDF events to DETAIL More overhead because of additional events traced.

Display Level Options

There are many levels of trace displays available. Each presents summarized and integrated information from the data collection level active for that trace request as shown in the display level summary below. In this summary, the data collection levels where specific data displays become available are shown in parentheses. For example, (S) indicates that the data is already collected by a summary trace request. (D) indicates a detail trace request; it includes data identified with (S).

- LTRAC - (S) Scrollable list of trace entries
- TSTAT - Trace statistics on all or selected trace entries

(S)	Trace status
(S)	Summary of key average measurements
(D,SCAN)	Database scan summary
(D,LOCK)	Database lock summary
(D,IO)	Database I/O summary
(D)	Sort summary

- TSUMA - Trace summary by AUTHID

(S)	Accounting data

- TSUMC - Trace summary by connection name

(S)	Accounting data

- TSUMP - Trace summary by plan

(S)	Accounting data

- TSUMT - Trace summary by time

(S)	Accounting data

- TSUML - Trace summary by location

(S)	Accounting data

- STRAC - Information about one trace entry

(S)	Accounting data
(D)	SQL statement summary by program
(D,SCAN)	Database scan summary
(D,LOCK)	Database lock summary
(D,IO)	Database I/O summary
(D)	Sort summary

- DTRAC - Events for one trace entry

(D)	LEVEL=1 - SQL
(D)	LEVEL=2 - SQL + DB2 requests
(D)	LEVEL=3 - SQL + DB2 requests + DB2 processing

- DTRAC,PGM=

  Events for one program (DBRM) within one trace entry

- Pop-ups - One event

(D)	Detail of one event

Usage Examples

This section describes a few of the possible uses of application traces. Its purpose is to help you discover how to use a trace to solve unique problems.

Application trace allows

- Application developers to test not only application correctness but performance as well.
- Database administrators to review SQL before accepting it for production use.
- Database administrators to analyze I/O by page set (table spaces and index spaces).
- Periodic performance reviews of specific application groups.
- Investigation of a specific plan suspected of poor performance (or identified in previous summary analysis).
- Analysis of index usage and database access paths (scans) and I/O and lock request patterns as necessary.
- Traces of QMF and ad hoc SQL usage to find users that need training in SQL concepts to improve performance.
- Ad hoc analysis of workload history for a DB2 system.
- Checks of the performance characteristics of a new application accessing production databases.
- Ongoing exception traces per workload type to be set up to detect bad applications that
  - Do not meet performance criteria or service levels
  - Affect total system performance

The next sections describe application trace techniques that site personnel can use.

(Application Developer) Test Application Performance

Application developers can be given the authority to run detail traces only on their test or development DB2 system where trace overhead is not critical. All types of tracing allowed can be specified for each user, SUMMARY, DETAIL, or ALL, with each additional type causing progressively more overhead.

Sample Trace Request Options

Option	Use
PARM=RBTEST3	Identify this test run with user initials and sequence number
TYPE=DETAIL	Relatively low overhead (panel default includes SQL)
START=	Leave blank to start immediately
STOP=30	Leave default trace duration of 30 minutes
WRAP=YES	Keep the latest occurrences
DB2AUTH=ABG01	Select own DB2 work by AUTHID
STORAGE=1000K	Trace display buffer space is in BBI-SS PAS extended private area
LOGTRAC=Y	Log to a single dynamically allocated data set
TITLE=‘TEST 3’	User-specified title to identify log data set

A detail trace answers a great many questions with relatively low system overhead. It selectively traces the key events in the life of a query or transaction, including all SQL. For example, sorts invoked, elapsed and CPU times, and the row statistics showing activity per SQL statement can identify problem areas quickly. It also includes EXPLAIN data for dynamic SQL, QMF queries, and binds of static plans. In addition, multiple index access path processing can be analyzed both when it is successful and when it fails.

Application developers now have an easy way of testing different versions of SQL and comparing the results. All the necessary information for every test iteration can be gathered in one trace showing SQL text, EXPLAIN data, and actual execution performance. The data can be reviewed online during the testing, recalled online from the log at a later time, or printed for more thorough analysis. A series of tests can be made over several days (for example, identified as TEST1, TEST2, and so on) and recalled online or printed for performance comparisons.

More data can be collected with the higher levels of detail trace (TYPE=D,SQL|SCAN|IO|LOCK|DDF|DDFVTAM):

- SCAN is needed occasionally to pick up the scans and show specific table and index usage.
- IO may be requested less frequently to trace the I/O patterns. However, this is the area most subject to change between test and production since the data access and buffering varies greatly with table size and the amount of concurrent activity.
- LOCK is needed only to check out locking patterns.
- DDF is needed to trace distributed SQL, either as requestor or server, in this location.
- DDFVTAM is needed to analyze the VTAM-related activity caused by distributed SQL.

A summary trace can be run after the SQL statements are checked. This trace provides elapsed and CPU times that are not artificially increased by the DB2 performance trace overhead.

When the occurrences that need to be evaluated are collected, the trace (and its overhead) can be stopped with a Z line command. The data from stopped traces can be browsed until the request is purged, or WRAP can be changed to NO with a modify trace request. This prevents previously collected data from being overwritten until it can be analyzed. Tracing can then be restarted with a modify trace request to change WRAP to YES. However, tracing overhead continues during this whole period.

When multiple programs are part of one plan, activity by program can be analyzed with the STRAC display. It presents an SQL summary by program for the total plan, with percent of total calculations of elapsed and CPU times per SQL statement and program. This immediately points out where the most time is being spent. Also, in the event traces (shown in the DTRAC displays), an identifier is appended to each SQL statement number if it is in a different program than the previous SQL. This shows the logical flow between programs. By expanding from one statement in the SQL summary to DTRAC, you can view just the detail events from that one program (DBRM).

Since this trace was logged, any of the online displays can be printed for offline analysis. You can submit a print job for this log directly from the HISTORY TRACES option or by tailoring and submitting a batch job.

(DBA) Review SQL Before Production Use

The following trace gives the DBA a simple way to check out the performance characteristics of any new plan:

Sample Trace Request Options

Option	Use
TYPE=D,SQL,SCAN	Medium overhead
START=	Leave blank to start immediately
STOP=	Leave blank to stop manually
WRAP=YES	Keep the latest occurrences in the buffer
DB2PLAN=XYZAB	Select the new plan
STORAGE=1000K	Trace display buffer space is in BBI-SS PAS extended private area
LOGTRAC=Y	Log to a dynamically allocated data set

By tracing scan information integrated with the SQL statements, critical performance indicators can be easily identified, such as sorting, index usage, scan patterns, work (DSNDB07) scans, and so on. Events and resources used can be seen directly as they actually happened. Bad SQL statements can be identified and fixed before they are put into production.

For large plans with many SQL statements, the SQL summary section of STRAC can be selected and sorted by resource usage to easily detect the worst offenders. From this section, it is possible to select a statement to view row statistics, or a program to view only the detail events for that single program within a plan.

Table space usage can be analyzed for the whole program by selecting the database summary section of STRAC, where you can view data, work (sort), and index usage of sort by any column to identify high usage. TSTAT offers the same display but for all traced threads.

To see I/O usage by table space, also include IO in the trace TYPE options and then browse the STRAC or TSTAT Database Lock and I/O Summary section. Use the DTRAC detail event display to correlate scan and I/O data to specific SQL statements.

An additional valuable analysis tool can be gained by also tracing the BIND of the plan. Specify DB2PLAN=XYZAB, BINDCT++, and start the trace before the bind. This provides a record of the text of each static SQL statement and its related EXPLAIN data. It is not necessary to specify the EXPLAIN option on the BIND. Dynamic SQL text and EXPLAIN data are included in the plan execution trace, giving a complete picture.

(DBA) Review Application Performance Periodically

The following trace can be used to review the summary data of all the traced transactions with the TSTAT display to determine if performance meets your expectations:

Sample Trace Request Options

Option	Use
TYPE=SUMMARY	Low overhead
START=11:00:00	Time period with activity
STOP=11:30:00	Just long enough
WRAP=NO	Fill up the buffer and stop
DB2PLAN=SALES+	Pick out all the sales applications
DB2CONN=IMSP	Only those executing from IMS production
STORAGE=100K	Buffer space for approximately 160 accounting records

The average values for elapsed time, CPU, number of SQL statements executed, and number of GETPAGEs should be enough to tell you if the standards for this type of workload are being met. For example, the number of SQL statements expected from an IMS or CICS transaction may be subject to site restrictions. Also, an unusually high number of GETPAGEs could indicate poor access paths.

Review whether the maximums of these values vary greatly from the averages, which can indicate occasional problems or a specific plan that is performing poorly. In this case, use the LTRAC scrollable display to look for the bad performers and select the STRAC accounting information for further detail. Before looking at STRAC details, you may want to use the trace summaries (TSUMA, TSUMC, TSUMP, TSUMT, TSUML) to summarize, sort, and select the LTRAC entries for ad hoc analysis. For example, select TSUMA to summarize by AUTHID and sort those entries to find the highest CPU user, and then select that line to return to an LTRAC list of only the entries for that user. These entries then could be summarized again by plan, connection name, or time. This process can be used iteratively to detect entries where further detailed analysis of STRAC accounting information is worthwhile.

Also consider the buffer pool indicators on the TSTAT display (average page updates and the ratio of GETPAGEs per READ I/O). This ratio should generally be approximately 2:1 for transaction workloads and higher for batch. Since these values are also broken out by buffer pool, unexpected use of the 32K pool is revealed.

(DBA) Investigate Problem Plans

If a specific plan is suspected of poor performance or identified in previous summary trace analysis, a detail trace can be set up for the next time that plan is expected to be run. Logging allows all the detail events to be traced, even for long-running batch jobs, without loss of data due to online buffer sizes. After the trace is complete, select this log data set from the History Traces application (Option 5), or select it as a current trace (Option 4), and use the HISTORY EXPAND button to view the logged data. The same analysis techniques can then be used as described previously in the DBA SQL review.

Sample Trace Request Options

Option	Use
TYPE=D,SQL,IO,SCAN	More overhead
START=	Leave blank to start immediately
STOP=20	Stop after 20 minutes
WRAP=YES	Keep the latest occurrences in the buffer
DB2PLAN=BIGGUY	Select the problem plan
DB2CONN=BATCH	Select batch only
STORAGE=500K	Trace display buffer space is in BBI-SS PAS extended private area
LOGTRAC=Y	Log the trace
TRCYL=20	Allocate a larger log to hold the long-running batch jobs

(DBA) Analyze Index Usage and Database Access Paths

The following trace can be useful for evaluating the effect of changes made to the tables to improve performance, for example, after a REORG, when a new index is added, or RUNSTATS is updated. When one or more plans are rebound, their new performance characteristics can be checked out with the trace. A detail trace plus scan events is needed to see the index usage. Additional I/O events would be necessary to analyze READ I/O.

Sample Trace Request Options

Option	Use
TYPE=D,SCAN	Medium overhead
START=	Leave blank to start immediately
STOP=60	Stop after 60 minutes
WRAP=NO	Fill up the buffer and stop
DB2PLAN=ABC,ADE	Select specific plans
STORAGE=500K	Trace display buffer space is in BBI-SS PAS extended private area

By logging a trace both before and after a REORG, the performance statistics of each can be compared, either by recalling both online in split screen mode, or by printing selected reports.

(DBA) Trace QMF and Ad Hoc SQL Usage

The following trace request provides the SQL statement text, EXPLAIN data, and the row statistics of the QMF queries. Only long-running queries are kept for later analysis.

Sample Trace Request Options

Option	Use
TYPE=DETAIL	Relatively low overhead
START=09:00:00	Cover main morning activity
STOP=11:30:00	Stop at lunch time
WRAP=YES	Keep the latest occurrences in the buffer
DB2PLAN=QMF310	Select QMF
DB2CONN=TSO	Select TSO access
STORAGE=1000K	Trace Display buffer space is in BBI-SS PAS extended private area
ELAP=12.5	Keep only those with elapsed time >=12.5 seconds
LOGTRAC=Y	Logging
TITLE=‘LONG QMF QUERIES’	Trace title

(DBA) Analysis of Short-Term Workload History

It is recommended that a summary trace for the total system be run continuously to capture accounting data for all transactions, queries, and batch jobs executed within the latest time period of the user’s choice. This request should be defined to start automatically. A default THRDHIST summary trace is included in BLKDMRW, but additional options can be specified. For example, specify logging to a series of seven permanently allocated data sets and a switch time of midnight. Automatic archiving can be set up to save the data past the one-week period covered.

The trace data is kept in BBI-SS PAS extended private storage so a STORAGE buffer size of one to four megabytes or higher is viable. One megabyte holds approximately 1,700 records. The system transaction rate determines the time period covered. For this type of trace, specify WRAP=YES so that the most current data is always available. With logging, earlier data can always be retrieved.

TYPE=SUMMARY	Low overhead
START=	Start immediately
STOP=	Run continuously
WRAP=YES	Keep the latest entries
STORAGE=4000K	Keep approximately 7,000 records
LOGTRAC=Y	Logging
TRNUMDS=7	Log data sets
TRSWTIME=24:00:00	Switch data sets at midnight
TRARCSTC=ARCJCLS1	Archive log data set when full
TRDISP=OLD	All data sets pre-allocated
TRDSN=‘DB2C.WEEKLY.SUM.V01’	First log data set DSN
TITLE=‘THREAD HISTORY’	Trace title
PARM=THRDHIST	Trace ID

The most current data is available by selecting Option 6, GRAPH, or the current trace THRDHIST from Option 4, TRACES. Use the HISTORY EXPAND button to see the activity of the whole day. Select any of the six previous days' history trace from Option 5, HISTORY TRACES.

To analyze the workload, select the appropriate trace summary by AUTHID, connection name, plan, location, or time from LTRAC. The summary displays show averages and totals of CPU, elapsed time, number of statements, and GETPAGEs. Sort by any of these columns (descending order) to find the highest resource users. Select any summary entry to display a subset of individual list entries with LTRAC.

The TSUMT summary by time defaults to 10-minute intervals, but this can be changed to any desired minute or hour interval. Selecting one of these entries provides an LTRAC list for only that time period. A desired time period also can be specified directly in the parameter field of LTRAC, TSTAT, or TSUMx.

To identify specific workload-related problems that occurred sometime earlier, first use the TSUMT summary by time to narrow down the scope of the search, and then the other summaries, selections, and sorting capabilities can help you identify the few records that require further analysis.

Once you have identified a problem thread, use the P line command against that log data set (Option 5) to generate and submit a job to print reports. For example, specify DB2AUTH to select only one user, and specify these options to print:

LTRAC=Y
STRAC=SUMMARY
TSTAT=Y
TSUM=P

(DBA) Analyze I/O by Page Set

It is occasionally useful to analyze I/O usage by database and table space/index space, either for a problem application or for the total DB2 system. This can point out DASD contention or poor application data access paths.

The need for I/O analysis is most easily detected from accounting records, either from current summary traces or from historical data in Performance Reporter to do trending. Excessive I/O times show up best with either Accounting Class 2 (In-DB2 elapsed time) or Class 3 (I/O wait elapsed time and number of events). This trace data then allows analysis down to the page set level.

Sample Trace Request Options

Option	**Use**
PARM=SYSIO	Trace identifier
TYPE=D,IO	Medium overhead
START=10:30:00	Cover peak period
STOP=30	Stop after 30 minutes
DB2AUTH=+	Trace all threads, including async I/O
LOGTRAC=Y	Log to a dynamically allocated data set
TITLE=‘SYSTEM I/O ANALYSIS’	Give trace a descriptive title

The collected I/O data can then be analyzed, either online or in printed reports. It is summarized by page set for the total DB2 system in the TSTAT display and for a single thread in the STRAC display. Each of these displays has a similar section, DATABASE LOCK AND I/O SUMMARY, that shows the number of synchronous reads and writes and the elapsed wait time, as well as the count of asynchronous reads and number of pages read.

The best way to analyze the I/O workload is to use the series of DBIOx displays, available with an expand button from LTRAC or TSTAT on an I/O trace. The first display (DBIO) summarizes the I/O events by database and table space. You can view synchronous I/O, asynchronous prefetch I/O, or total I/O. 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.

The **total** elapsed time identifies the impact of I/O to each page set as a component of application response time. An unusually high **average** elapsed time can point out possible poor DASD response time. In this case, check the DBTS display to see whether a large number of extents could be causing the problem, indicating the need for a REORG. If that is not the case, you may want to notify your MVS DASD analyst of a potential DASD contention problem.

At any time you can return to LTRAC and 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, either summarized in STRAC or as individual events in DTRAC.

If you identify an application problem, you may want to start a different trace that includes SQL, so that you can relate the I/O events to specific SQL statements. From this type of I/O trace, there are two additional DBIOx displays that summarize the I/O events by plan and package/program, or even by SQL statement.

However, be aware that if you qualify a trace by plan, a DB2 limitation means that you will not be able to see the prefetch reads, only the synchronous reads and writes done under the application thread. To see asynchronous I/O, you can qualify by one or more authids as long as you add SYSOPR to the list.

(DBA) Analyze Lock Contention

The first place to look is the Lockout History display that shows you a list of the last 100 timeouts and deadlocks that have occurred. The complete details from these events are available to identify both holder and waiter threads as well as the resource(s) involved.

To analyze locking problems for an application, first look at the accounting record (STRAC) for the counts on maximum page locks, suspensions, lock requests, and timeout/deadlocks. This data is already available with a summary trace. This should help you identify which applications are having—or causing—problems. For further information, a detail trace with SQL events is often sufficient. You usually will want to include SQL so you can easily determine which statement is causing the problem, but a great deal of information is available with just the basic detail trace.

If you are still searching for the culprit application, you may want to run a basic detail trace with no further events (not even SQL), but covering all the suspected applications. This type of trace causes much less overhead. You could also use exception filters in the trace request to identify threads with lock suspensions, escalations, lock table requests, or those with a high maximum number of locks held.

The following events are available in DTRAC for any detail trace. Analyze this information before deciding to trace LOCK events.

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 TIMEOUT event (display LEVEL=2) that gives complete details.

Note: LOCK-TMO is not available for DB2 4.1.

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.

Another source of information is the DB2 timeout and deadlock messages in the Journal Log or MVS console. If the DB2 messages are being written to the Journal, go to Option L and issue a FIND DSNT37 to find any deadlock (DSNT375) or timeout (DSNT376) messages. They are followed by DSNT501I to identify the resource and lock owner that caused the contention. If you have AutoOperator Solutions installed, you can use the DB2DLOK solution to summarize and analyze the deadlock/timeout messages and LOCKD contention snapshots. Refer to the *MAINVIEW Solutions Guide* for details.

Note: A detail trace including LOCKS should be run only for a selected thread because of the high volume of lock events.

(System Programmer) Check New Production Application Performance

Sample Trace Request Options

Option	Use
TYPE=SUMMARY	Low overhead
START=	Leave blank to start immediately
STOP=	Leave blank to stop manually
WRAP=NO	Fill up the buffer and stop
DB2PLAN=XYZAB	Select the new plan
STORAGE=100K	Trace display buffer space is in BBI-SS PAS extended private area

Authority to run traces in production systems, especially detail traces, should be restricted to a few DBAs or systems programmers. This limits the amount of overhead that could be suddenly put on a production system, thus reducing performance problems. Someone needs the authority, because application performance can vary drastically between test and production mainly because of the potential differences in table sizes.

Each new application should be checked during its first days in production. This is done by running a summary trace and reviewing the averages shown in TSTAT and TSUMx. If these averages (or unusually high maximums) indicate a problem, a detail trace can be run for a short period to gather more information on a few plan executions.

A good procedure is to run a summary trace in the test system just before cutover so that you have benchmark values to compare to your production results. Summary trace records can be logged and printed for later analysis if this is preferred.

One important feature about trace is that it can be run at any time. The application developer does not need to be notified and a plan REBIND that could potentially affect performance does not need to be done.

(System Programmer) Set Up Ongoing Exception Traces per Workload Type

Sample Trace Request Options

Option	Use
TYPE=SUMMARY	Low overhead
START=	Leave blank to start immediately
STOP=	Leave blank to run continuously
WRAP=YES	Keep the latest occurrences
LOGTRAC=YES	Log the records for batch analysis
DB2CONN=CICS+	Select all CICS transactions
STORAGE=300K	Space for approximately 500 records
ELAP=2.5	Keep if elapsed time >= 2.5 seconds
ABORT=Y	Keep if aborted
UIDCOM=10	Keep if UPDATEs+INSERTs+DELETEs >= 10
GETRIO<1.5	Keep if GETS per READIO < 1.5
GETPG=500	Keep if GETPAGEs >= 500
MAXLOCK=20	Keep if maximum held locks >=20
SQLTOT=25	Keep if number of SQL statements >= 25

Exception traces per workload type (IMS, CICS, TSO, BATCH) can be set up to run continually. The Exception Filters vary by workload type. The filters are processed with OR logic so that any record with at least one of the specified conditions can be retained and logged.

These requests should be run the first few days under observation to see whether the filters are set correctly to select only real exceptions that require later analysis. After the exception filter values are tuned, these requests should be added to the block of requests that are started automatically when the BBI-SS PAS starts. (See BLKDMRW in BBPARM.) When the target DB2 is active, the traces are activated automatically.

When questions arise during the day about application performance, these exception records can be browsed for information. The LTRAC display shows each occurrence in a scrollable list with the timestamp when it completed. The STRAC display for any one of these occurrences can be viewed by cursor selection. This provides most of the DB2 accounting information. The value that caused a filter to be activated is shown in the key indicators section.

The records on the log can be used to create various summary reports. The offline printouts can be analyzed daily to isolate problem applications before performance degrades too far (for example, because of the need for a REORG).

Chapter 9. Application Trace Displays

All trace displays described in this section are available for both current traces where the data is still in online buffers, and history traces that have been saved in the trace log data sets. The first display shown (LTRAC) provides access to all other trace displays through EXPAND or line selection.

A current application trace can be viewed by:

- Selecting an active or complete trace (if still active, only completed transactions are shown):
  1. Select the list of current traces using Option 4.
  2. Use the S line command to select the LTRAC display of trace entries in the online buffers for one trace.

A historical application trace can be viewed by:

1. Selecting the History Traces application (Option 5).
2. Using the S line command to select the LTRAC display of trace entries in a trace log data set.

A currently active thread that is being traced can be viewed by:

- EXPANDing from the DUSER display to view the trace events (UTRAC display).
  - Position the cursor on UTRAC in the EXPAND line of the DUSER display or on any line in the body of the display and press ENTER to view all the traced events of the current active thread for a specified user (see "UTRAC—User Detail Trace" on page 380).

    If a trace is active for that thread, the DUSER resource analyzer display shows an EXPAND FOR DETAIL TRACE message.

Trace Display Summary

The trace displays comprise the following:

- LTRAC provides a scrollable list of trace entries. Each entry corresponds to a completed DB2 transaction or thread (DB2 accounting record).
- TSTAT provides:
  - A summary of all or selected records in the trace buffer and trace status.

    The average, maximum, minimum, and total for the key measurements of CPU, elapsed time, number of statements, and GETPAGEs are summarized. This gives a good indication of whether additional checking is necessary.

- Database and sort summary for all or selected records (detail trace only).

  With a detail trace, a database summary can indicate unexpected table space scans or index usage. It shows the number of scans and number of pages scanned per database object which is broken down by type (index, data, or work). Synchronous reads and writes and lock suspensions are also shown if traced.

  Since DB2 sorting can cause unexpected overhead, a sort summary provides information on the number of sorts, number of records sorted, and the sort elapsed time and record size.

- TSUMA provides a scrollable list of all or selected trace entries summarized by AUTHID.
- TSUMC provides a scrollable list of all or selected trace entries summarized by connection name.
- TSUMP provides a scrollable list of all or selected trace entries summarized by plan.
- TSUMT provides a scrollable list of all or selected trace entries summarized by time.
- TSUML provides a scrollable list of all or selected trace entries summarized by location.
- TSUMR provides a scrollable list of all or selected trace entries summarized by DB2 correlation ID.
- DBIOx provides a scrollable list of I/O activity and measurements from a detail I/O trace, summarized by:
  - DB/TS (DBIO)
  - AUTHID (DBIOA)
  - Buffer pool (DBIOF)
  - Connection (DBIOC)
  - Time (DBIOT)
  - Location (DBIOL)
  - Plan (DBIOP)
  - Plan and package (DBIOK) (only with SQL trace)
  - Plan, package, and SQL statement (DBIOS) (only with SQL trace)
- STRAC provides summary information for one transaction that includes:
  - A display of information from the accounting record, including elapsed time and buffer pool usage analysis sections.
  - DDF activity summary.

    The DDF summary displays statistics for each remote location associated with the thread.
  - SQL statements per program for one transaction (detail trace only).

    At the transaction level, the detail trace provides an SQL statement summary per unique program. Information on elapsed CPU times, sorts, and scans (index, data, and work) per call statement identify statements or programs that require further analysis.

    Note: Select the SQL expand option to be able to sort the statements by resource usage.

- Database summary for the single transaction (detail trace only).

  The database summary provides the same information as with the TSTAT display only for the single transaction. This section can be viewed separately (SCANS) and sorted by these resource values to make analysis of large plans simpler.

- Database lock and I/O summary for the single transaction (detail trace only).

  The database lock and I/O summary provides the same information as with the TSTAT display only for the single transaction. This section can be viewed separately (IO/LOCK) and sorted by these resource values to make analysis of large plans simpler.

- Sort summary (detail trace only).

• DTRAC and UTRAC provide:

  - A scrollable list of trace events (detail trace only) that can be viewed at three levels:

    LEVEL=1
    User requests (SQL only)

    LEVEL=2
    Level 1 plus DB2 requests, such as CREATE THREAD (default)

    LEVEL=3
    Levels 1 and 2 plus DB2 processing, such as scans and sorts

    Note: A parameter of PGM=xxx limits the display to events from one program.

  - Pop-up displays that expand event detail (detail trace only).

    Each event line contains elapsed and CPU times plus additional information to aid in problem determination. The additional information, such as row statistics per statement or scan, can be viewed with a pop-up display for that event.

Navigating Between Trace Displays

You can use the following techniques to move from one display to another:

• EXPAND

  This feature provides access to the next level of detail. Any display with an EXPAND selection bar can be expanded. For trace displays, you can use it to move from:

  - LTRAC (DB2 Trace Entries) to TOTALS (TSTAT Trace Statistics), AUTHID, CONNECT, PLAN, TIME, or LOCATION (TSUMx Trace Summaries) by cursor selection of the corresponding EXPAND bar button to see a summary of the traced transactions.

  - TSUMx to LTRAC by cursor selection of one of the summary lines to see a subset of the traced transactions.

  - LTRAC to STRAC (Summary Trace Entry) for a single entry by cursor selection on an LTRAC entry to see the accounting data (and detail data summaries) for one transaction.

- STRAC to a specific section of the STRAC display by cursor selection of the second line of the EXPAND bar to see these sections:
  - SQL Summary (sort available)
  - Database Summary (sort available)
  - Database Lock and I/O Summary
  - Sort Summary
- STRAC to DTRAC by cursor selection of the DETAIL EXPAND bar button to see the detail chronological events.
- STRAC to DTRAC for one program by cursor position within the SQL section of the STRAC or STRAC(SQL) display to see only the events for one program (DBRM).
- DTRAC or UTRAC to the pop-up for any event marked with an asterisk by cursor selection on the marked DTRAC entry to see all the details from the IFCID for that event.

  There are three DTRAC and UTRAC display levels available (LEVEL = 1|2|3) for an active detail trace request. Each level shows additional events. Overtype the value in the LEVEL=n parameter field to change levels.

- PF7/8

  Use PF7 and PF8 to scroll up and down in any of the following scrollable displays:

LTRAC	List of trace entries
TSTAT	Trace status and trace summaries
TSUMA	Trace summary by AUTHID
TSUMC	Trace summary by connection name
TSUMP	Trace summary by plan
TSUMT	Trace summary by time
TSUML	Trace summary by location
TSUMR	Trace summary by correlation ID
STRAC	Accounting data and detail summaries for one transaction/thread
DTRAC	List of events
Pop-ups	Any display longer than the window size

- PF10/11

Use PF10 and PF11 to move to previous or next entries of a like kind. This includes:

STRAC	One trace entry (transaction) to the next
DTRAC	One trace entry (transaction) to the next
Pop-ups	One pop-up to the next within a DTRAC or UTRAC event list

Figure 27 on page 113 shows graphically how you can navigate from one trace display to another.

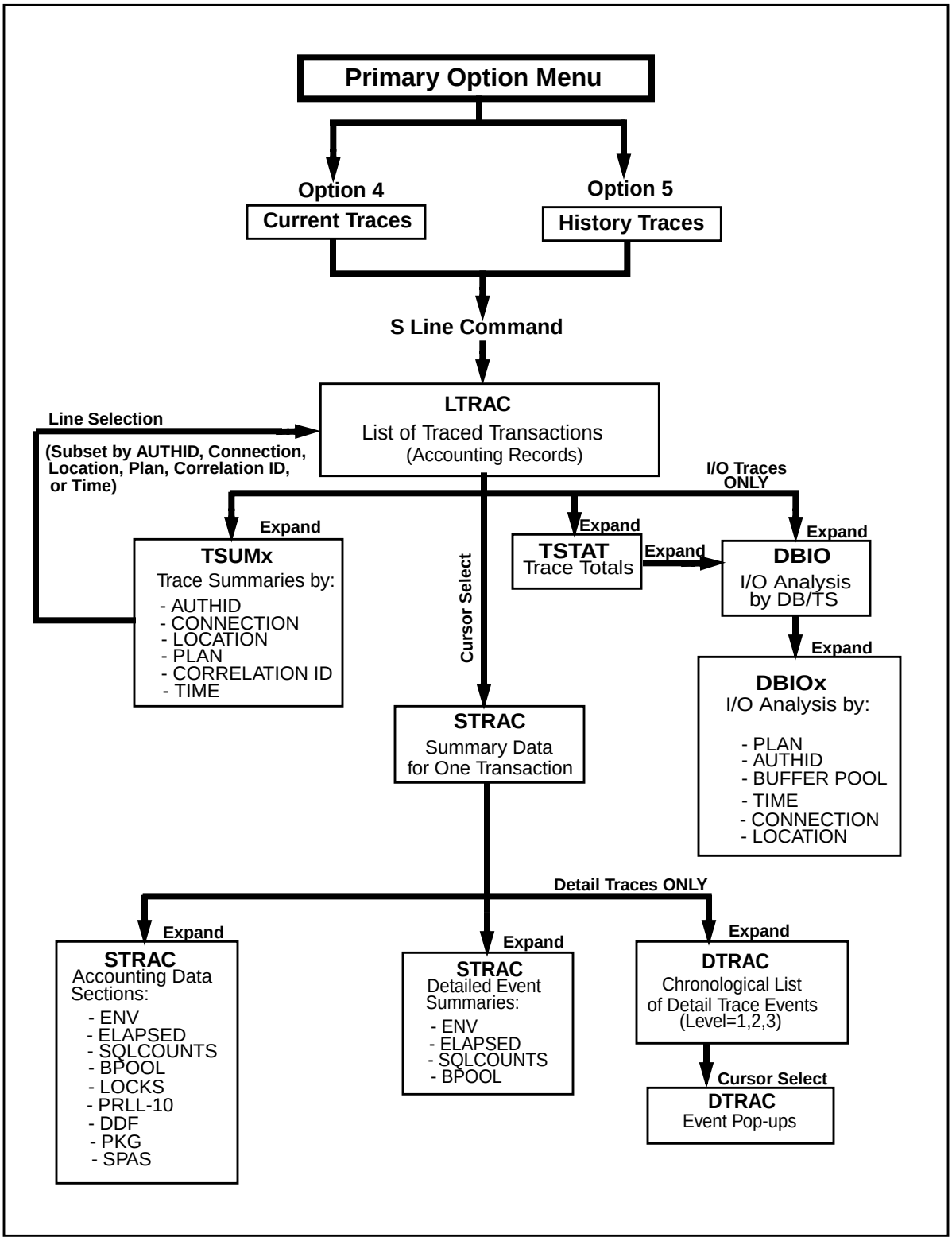

Figure 27. Trace Display Service Access

Chapter 10. Viewing a List of Completed Threads

To view a list of completed threads (LTRAC display), use the S line command to select a trace from either the list of Current Traces (Option 4) or the list of History Traces (Option 5).

LTRAC—DB2 Trace Entries

The DB2 Trace Entries panel is a scrollable list of trace entries currently in the trace buffer.

```
BMC Software --------------   DB2 TRACE ENTRIES     -------------  RX AVAILABLE
SERV ==> LTRAC           INPUT    14:11:09  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02                                      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, CORR, LINESEL(STRAC)
  19MAR01                  PBCRCP - SQL/SCAN/IO
  END TIME     PLAN    AUTHID  CONNECT   ELAPSED   CPU    # STMTS GETPAGE REASON
 ----------- -------- -------- -------- -------- -------- ------- ------- ------
 09:17:52.76 DSNTIA21 CIR8X    BATCH       13 s    268 ms       4     459 OK TRM
 09:18:10.33 DSNTIA21 CIR8X    BATCH       11 s    220 ms      12     168 OK TRM
 09:18:32.78 DSNTIA21 CIR8X    BATCH    8,768 ms   218 ms      12     162 OK TRM
 09:18:55.13 DSNTIA21 CIR8X    BATCH    9,225 ms   214 ms      12     162 OK TRM
 09:19:19.80 DSNTIA21 CIR8X    BATCH       11 s    216 ms      12     162 OK TRM
 09:19:44.59 DSNTIA21 CIR8X    BATCH       11 s    216 ms      12     162 OK TRM
 09:20:09.28 DSNTIA21 CIR8X    BATCH       11 s    217 ms      12     162 OK TRM
 09:20:34.83 DSNTIA21 CIR8X    BATCH       11 s    215 ms      12     162 OK TRM
 09:21:00.54 DSNTIA21 CIR8X    BATCH       12 s    218 ms      12     162 OK TRM
 09:21:24.18 DSNTIA21 CIR8X    BATCH       10 s    215 ms      12     162 OK TRM
 09:21:47.62 DSNTIA21 CIR8X    BATCH       10 s    217 ms      12     162 OK TRM
 09:22:05.21 DSNTIB21 CIR8X    BATCH    1,117 ms   112 ms      49      59 OK TRM
 09:22:16.03 DSNUTIL  CIR8X    UTILITY  2,907 ms   163 ms       0      64 OK TRM
 09:22:17.93 DSNUTIL  CIR8X    UTILITY  1,886 ms   107 ms       0      61 OK TRM
 09:22:36.33 DSNTIA21 CIR8X    BATCH       10 s    204 ms       4     357 OK TRM
 09:22:43.14 DSNTIB21 CIR8X    BATCH    1,981 ms    73 ms      49      32 OK TRM
 09:22:59.79 DSNTIB21 CIR8X    BATCH    1,439 ms    69 ms      49      30 OK TRM
```

Figure 28. DB2 Trace Entries Panel

Select Code:

LTRAC

Parameter:

The following parameters can be specified:

trace id

Unique ID identifying the trace. A blank can be an identifier. Therefore, this parameter is positional and a comma delimits the next parameter. For example, if there is no identifier, a comma must precede any other parameter. However, if there is an ID, it must be in the first position.

,A|AUTHID=xxxxxxxx

Selects a subset of the trace entries by AUTHID. Specify one AUTHID, or a generic group of AUTHIDs by replacing character positions with a + (one position) or an * (multiple positions).

,C|CONNECT=xxxxxxxx

Selects a subset of the trace entries by connection name. Specify one connect type (TSO, BATCH, DB2CALL, imsid, cicsjobname), or a generic group of connections by replacing character positions with a + (one position) or an * (multiple positions).

,P|PLAN=xxxxxxxx

Selects a subset of the trace entries by plan. Specify one plan, or a generic group of plans by replacing character positions with a + (one position) or an * (multiple positions).

,L|LOC=xxxxxxxx

Selects a subset of the trace entries by location. Specify one location, or a generic group of locations by replacing character positions with a + (one position) or an * (multiple positions).

,R|CORRID=xxxxxxxxxxxx

Selects a subset of the trace entries by correlation ID. Specify one correlation ID, or a generic group of correlation IDs by replacing character positions with a + (one position) or an * (multiple positions).

,T|TIME=hhmm[-hhmm]

Selects a subset of the trace entries by a start time or time period, where hhmm indicates hours and minutes.

Midnight wrap is supported where the start time is higher than the end time; for example, TIME=2000-3000. The end timestamp of an accounting record is used to assign it a time period.

,D|DAY=nn[-nn]

Selects a subset of the trace entries by a single day or range of days, where nn can be a relative number from the date of the first accounting record in the trace buffer (first date = 1, up to a maximum of 99).

If a time period is selected without a DAY specification, it defaults to the day of the most current record in the buffer.

If there is data for only one day in the trace buffer, the format of the timestamp is hh:mm:ss. If there is data for more than one day, the format is dd-hh:mm, where dd is the relative day number from the date of the first accounting record in the trace buffer.

Each accounting record traced is assigned to a time period based on its end timestamp. Therefore, the processing done for that unit of work is assigned to one interval, but it may have partially occurred during one or more previous time intervals depending on its elapsed time.

Note: Each keyword can be specified only once. If more than one selection keyword is specified, the result is ANDed.

Description:

Provides a scrollable list of all the active entries for the selected trace. There is an entry for each DB2 accounting record. A record is written for each DB2 transaction and is available to the trace facility when the transaction terminates. For more information, see the IBM publication, *DB2 Administration Guide*.

Note: If the target for a historical trace is not equal to the current TGT field, it is displayed in a HIST TGT field.

An asterisk (*) is used in this display to indicate the following circumstances:

- An asterisk preceding the PLAN column indicates this is a parallel query or utility subtask.
- An asterisk preceding the CONNECT column indicates this is a Database Access Thread (DBAT).

Scrolling:

CSR in the SCROLL field indicates the display is scrollable. Active entries are copied to a separate storage area in the BBI-SS PAS extended private area when this service is first invoked. Thus, you can scroll the data and it does not change. This scrollable buffer is refreshed whenever ENTER is pressed in the LTRAC service and no line is selected by cursor placement.

If the number of entries do not fit on one screen, the list can be scrolled as described in “Scrolling a Display” in *Using MAINVIEW.*

Service Message(s):

A message in the parameter field of the service shows the sequence number of the first trace entry displayed and the total number of entries that can be displayed in a scrollable list. A message just below it shows the sequence number of the first and last trace entries currently available in the buffer. When displaying a history trace, this is the number of trace entries in that data set.

Note: These sequence numbers are internally assigned, in sequence, to each trace entry from the start of tracing. If the entries have wrapped in the buffer (WRAP=YES) and have overlaid earlier entries, the first sequence number in the buffer is no longer 1.

Expand:

The LTRAC display can be EXPANDed to the following displays:

MON(WKLD)
: Active Timer Requests display of all active monitors in the DB2 workload (WKLD) area.

TOTALS
: DB2 Trace Statistics display (TSTAT) for this trace.

I/O-DB/TS
: Display of I/O analysis by database and table space (DBIO) for this trace (available for an I/O detail trace only).

HISTORY
: Show the data from the current trace log data set for this trace (if you are viewing a current trace with logging, this provides access to data no longer in the online buffer). It is only shown when logging is active.

CURRENT
: The active trace data still in the buffers (if you are viewing a trace log data set for a still-active trace or a complete trace that has not yet been purged).

AUTH
: Trace Summary by AUTHID display (TSUMA) for this trace.

CONNECT
: Trace Summary by Connect display (TSUMC) for this trace.

PLAN
: Trace Summary by Plan display (TSUMP) for this trace.

TIME
: Trace Summary by Time (TSUMT) display for this trace (default of one-hour intervals).

LOC
: Trace Summary by Location display (TSUML) for this trace.

CORR
: Trace Summary by Correlation ID display (TSUMR) for this trace.

LINESEL(STRAC)
: A detailed display of information about a specific entry (STRAC) can be selected from this list by positioning the cursor anywhere in the line for that entry and pressing ENTER.

Area 1 - Trace Buffer Entry List

```
 19MAR01                  PBCRCP - SQL/SCAN/IO
 END TIME     PLAN    AUTHID  CONNECT   ELAPSED   CPU    # STMTS GETPAGE REASON
----------- -------- -------- -------- -------- -------- ------- ------- ------
09:17:52.76 DSNTIA21 CIR8X    BATCH       13 s    268 ms       4     459 OK TRM
09:18:10.33 DSNTIA21 CIR8X    BATCH       11 s    220 ms      12     168 OK TRM
09:18:32.78 DSNTIA21 CIR8X    BATCH    8,768 ms   218 ms      12     162 OK TRM
09:18:55.13 DSNTIA21 CIR8X    BATCH    9,225 ms   214 ms      12     162 OK TRM
09:19:19.80 DSNTIA21 CIR8X    BATCH       11 s    216 ms      12     162 OK TRM
09:19:44.59 DSNTIA21 CIR8X    BATCH       11 s    216 ms      12     162 OK TRM
09:20:09.28 DSNTIA21 CIR8X    BATCH       11 s    217 ms      12     162 OK TRM
09:20:34.83 DSNTIA21 CIR8X    BATCH       11 s    215 ms      12     162 OK TRM
09:21:00.54 DSNTIA21 CIR8X    BATCH       12 s    218 ms      12     162 OK TRM
09:21:24.18 DSNTIA21 CIR8X    BATCH       10 s    215 ms      12     162 OK TRM
09:21:47.62 DSNTIA21 CIR8X    BATCH       10 s    217 ms      12     162 OK TRM
09:22:05.21 DSNTIB21 CIR8X    BATCH    1,117 ms   112 ms      49      59 OK TRM
09:22:16.03 DSNUTIL  CIR8X    UTILITY  2,907 ms   163 ms       0      64 OK TRM
09:22:17.93 DSNUTIL  CIR8X    UTILITY  1,886 ms   107 ms       0      61 OK TRM
09:22:36.33 DSNTIA21 CIR8X    BATCH       10 s    204 ms       4     357 OK TRM
09:22:43.14 DSNTIB21 CIR8X    BATCH    1,981 ms    73 ms      49      32 OK TRM
09:22:59.79 DSNTIB21 CIR8X    BATCH    1,439 ms    69 ms      49      30 OK TRM
```

This area shows trace entries currently in the trace buffer by the following:

END TIME
: Time the transaction ended. The date displayed above END TIME is the date of the first data line shown.

PLAN
: Plan name the transaction executed.

Note: An asterisk (*) at the beginning of this column indicates this is a parallel CPU subtask.

AUTH ID
: Authorization ID.

CONNECT
: Connection name as:

- TSO
- BATCH
- DB2CALL (Call Attach Facility)
- IMS Subsystem ID
- CICS jobname

Note: An asterisk (*) at the beginning of this column indicates this is a Database Access Thread (DBAT).

ELAPSED
: Elapsed time as recorded in the DB2 accounting record. It adjusts to either seconds (s), milliseconds (ms), or microseconds (us) depending on the value. If the time exceeds 60 seconds, it is shown in the format hh:mm:ss.

CPU
: CPU time as recorded in the DB2 accounting record. It adjusts to either seconds (s), milliseconds (ms), or microseconds (us) depending on the value. CPU time includes all TCB and SRB time in all address spaces for processing the transaction, as seen by DB2.

STMTS
: Number of SQL statements issued by the transaction.

GETPAGE
: Number of GETPAGE requests issued by the transaction.

REASON
: The reason the transaction terminated can be one of the following:

  OK
  : Normal termination.

  AB TRM
  : Abnormal termination.

  R IN D
  : Resolve indoubt processing.

  FORCE
  : -STOP FORCE command issued.

  INCMP
  : The transaction has not yet terminated. Data is incomplete in the trace log data set being viewed. Only detail data is available; no DB2 accounting statistics exist.

Chapter 11. Viewing Summary Data for All Trace Entries

To view summary data for all trace entries, expand from LTRAC to the TSTAT display.

TSTAT—Trace Statistics

The Trace Statistics panel displays summarized statistics for the trace and all or selected transaction entries in the trace buffer.

Note: Summaries of detail events can be produced for detail traces, but this is done only when specifically requested with a parameter. Because of the amount of processing this may require, consider using the workload selection parameters first to limit the amount of data. The headings for detail summaries are displayed with no data if a detail trace does not select the events that make up the section (EVENTS NOT TRACED) or if there is no data to collect (NO EVENTS CAPTURED).

The following figures show representative displays of each TSTAT section:

```
BMC Software --------------    TRACE STATISTICS     -------------  RX AVAILABLE
SERV ==> TSTAT           INPUT    19:31:17  INTVL=> 5  LOG=> N  TGT==> DB2HHH
PARM ==> THRDHIST                                   ROW 1 OF 222 SCROLL=> CSR
    DZ3135W - USE ,DTL FOR DETAIL DATA                        HIST TGT--- DSNT
        ACCOUNTING: ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, SPAS, DDF
        SUMMARIES:  SCANS, IO/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
-- IFI 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 29. Summary Statistics for all Trace Buffer Entries (Base Section)

See “Summary Statistics (Base Section)” on page 130.

```
- - - - - - - - ELAPSED TIME ANALYSIS (ACCTG CLASSES 2,3 ONLY)- - - - - - - - -

CATEGORY           #EVENTS AVG/EVENT  ELAPSED  %TOTAL
----------------   ------- --------- -------- ------- 0 ...25...50...75..100%
ELAPSED TIME                                           |                      |
  IN DB2                12     20 s  00:03:58   38.07 | *******              |
  IN APPLICATION        12     32 s  00:06:27   61.92 | ************         |
 --TOTALS--             12     52 s  00:10:24  100.00 | ******************** |
WAITS IN DB2 (LOCAL)                                   |                      |
  LOCK/LATCH            11  3,148 ms    35 s     5.54 | *                    |
  I/O WAIT             160     17 ms 2,652 ms    0.42 | <                    |
  LOG WRITE I/O          2  5,151 us    10 ms    0.00 |                      |
  OTHER READ            15     42 ms   632 ms    0.10 | <                    |
  OTHER WRITE            0      0 us     0 us    0.00 |                      |
  UNIT SWITCH EVENTS                                   |                      |
  ..COMMIT/ROLLBK       34     72 ms 2,438 ms    0.39 | <                    |
  ..OPEN/CLOSE          90  1,632 ms 00:02:27   23.53 | ****                 |
  ..SYSLGRNG            35     38 ms 1,347 ms    0.21 | <                    |
  ..DATASPACE MGR       89    467 ms    42 s     6.65 | *                    |
  ..OTHER               37     26 ms   950 ms    0.15 | <                    |
  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--       220  1,050 ms 00:03:51   37.02 | *******              |
*NOT ACCOUNTED                      1,464 ms    0.23 | <                    |
********************************* END OF DATA *********************************
```

Figure 30. TSTAT Elapsed Time (ELAPSED) Section

See “Elapsed Time Analysis” on page 139.

```
- - - - - - - - - - - - SQL STATEMENT EXECUTION COUNTS - - - - - - - - - - - -
             TOTAL  AVERAGE        TOTAL   TOTAL
SELECT.........209      1.0   LOCK TABLE.............0   CREATE............4
INSERT.........155      0.8   GRANT/REVOKE...........7   DROP..............1
UPDATE.........219      1.1   SET CURR.SQLID.........7   ALTER.............0
DELETE..........87      0.4   SET HOST VAR. .........12
                              SET CURR.DEGREE........0   COMMENT ON........0
PREPARE........247      1.2   SET CURR. RULES........0   LABEL ON..........0
DESCRIBE........18      0.0   CONNECT TYPE 1.........0
                              CONNECT TYPE 2.........0   CREATE TEMP.......0
OPEN...........352      1.8   SET CONNECTION.........0   RENAME TBL........0
FETCH........2,248     11.7   RELEASE................0
CLOSE..........343      1.7   CALL...................0
ASSOCIATE LOCATOR.......0
ALLOCATE CURSOR.........0

** DML.......3,878     20.3   ** DCL................26   ** DDL............5
** REOPT.........0      0.0
                  - - - - DYNAMIC SQL CACHE TOTALS - - - -
STMT FOUND............0    KPDYN-PREP.AVOID.........0    STMT DISCRD........0
STMT NOT FOUND........0    KEEPDYN_IMPL.PREP........0    STMT PURGED........0
******************************** END OF DATA ********************************
```

Figure 31. TSTAT SQL Statement Execution Counts (SQLCOUNTS) Section

See “SQL Statement Execution Counts” on page 144.

```
- - - - - - - - - - - - - - BUFFER POOL ACTIVITY- - - - - - - - - - - - - - - - -
                      ----- TOTAL -----  ------- BP0 -----  ------- BP2 -----
ACTIVITY                 TOTAL  AVERAGE     TOTAL  AVERAGE     TOTAL  AVERAGE
                      -------- --------  -------- --------  -------- --------
GETPAGES               11,767    147.0    11,630    145.3       137      1.7
SYNC READS                 84      1.0        81      1.0         3      0.0
GETPAGES/READIO         140.0              143.5               45.6

COND GP FAILURES            0      0.0         0      0.0         0      0.0
SEQ PREFETCH REQS       1,383     17.2     1,372     17.1        11      0.1
LIST PREFETCH REQS          0      0.0         0      0.0         0      0.0
DYNAMIC PREFETCHS           0      0.0         0      0.0         0      0.0
ASYNC PAGES READ       10,583    132.2    10,562    132.0        21      0.2
PAGES/PREFETCH REQ        7.6                7.6                1.9

PAGE UPDATES                0      0.0         0      0.0         0      0.0
IMMEDIATE WRITES            0      0.0         0      0.0         0      0.0

HP SYNC READS             168      2.1       168      2.1         0      0.0
HP SYNC READ FAILS         10      0.1        10      0.1         0      0.0
HP ASYNC PAGES READ       250      3.1       250      3.1         0      0.0
HP WRITES                 249      3.1       249      3.1         0      0.0
HP WRITE FAILURES           0      0.0         0      0.0         0      0.0

            - - - GLOBAL BUFFER POOLS (DATA SHARING ONLY) - - -
ACTIVITY             ----- TOTAL -----  ------- BP0 -----  ------- BP1 -----
-------------------  -----------------  -----------------  -----------------
CF READS  (BUFFER CROSS INVALIDATION)
 -DATA RETURNED....         13      0.0        11      0.0         2      0.0
 -R/W INTEREST.....          6      0.0         6      0.0         0      0.0
CF READS  (DATA NOT IN BUFFER POOL)
 -DATA RETURNED....         62      0.3        62      0.3         0      0.0
 -R/W INTEREST.....        200      1.0       200      1.0         0      0.0
 -NO R/W INTEREST..          5      0.0         5      0.0         0      0.0
CF-WRITE (CHANGED)         160      0.8       127      0.6        18      0.0
CF-WRITE (CLEAN)            0      0.0         0      0.0         0      0.0
NBR UNREG PAGES...         50      0.2        50      0.2         0      0.0
NBR EXPLICIT XI...          0      0.0         0      0.0         0      0.0
NBR WRITES 2ND BP.          0      0.0         0      0.0         0      0.0
******************************** END OF DATA ********************************
```

Figure 32. TSTAT Buffer Pool (BPOOL) Section

See “Buffer Pool Activity” on page 148.

```
- - - - - - - - - - - - - - LOCK ACTIVITY - - - - - - - - - - - - - - - - - - -
                      TOTAL  AVERAGE                              TOTAL  AVERAGE
TIMEOUT..................2      0.0   MAX PAGE/ROW LOCKS..........109      1.3
DEADLOCK.................0      0.0
                                     LOCK REQUEST................548      6.8
SUSPEND-LOCK.............2      0.0   UNLOCK REQUEST..............434      5.4
SUSPEND-LATCH............1      0.0   QUERY REQUEST.................0      0.0
SUSPEND-OTHER............0      0.0   CHANGE REQUEST...............47      0.5
                                     OTHER REQUEST.................0      0.0
ESCALATION(SHR)..........0      0.0
ESCALATION(EXCL).........0      0.0

DRAIN REQUEST............0      0.0   DRAIN FAILURE.................0      0.0
CLAIM REQUEST..........132      1.6   CLAIM FAILURE.................0      0.0
                    - - - - GLOBAL LOCK ACTIVITY - - - -
LOCK REQUEST............10      0.1   XES LOCK REQUEST............234      2.9
LOCK CHANGE REQST........0      0.0   XES CHANGE REQUEST..........10      0.1
UNLOCK REQUEST...........0      0.0   XES UNLOCK REQUEST.........162      2.0

IRLM GLBL SUSPEND........0      0.0   XES GLOBAL SUSPEND............0      0.0
FALSE CONTENTION.........2      0.0

INCOMPAT. RETAINED.......0      0.0   NOTIFY MESSAGES SENT..........0      0.0
******************************** END OF DATA ********************************
```

Figure 33. TSTAT Locks (LOCKS) Section

See “Lock Activity” on page 151.

```
- - - - - - - - - - - - - - - - PARALLELISM - - - - - - - - - - - - - - - - - -
                          TOTAL  AVERAGE                                TOTAL
MAXIMUM DEGREE (MAX=3)  ......42     0.01   FALLBACK - NO BUFFER.............0
GROUPS EXECUTED...............21     0.00   FALLBACK - NO ESA SORT...........0
-- PLANNED DEGREE.............21     0.00   FALLBACK - AMBIG. CURSOR.........0
-- REDUCED DEGREE.............0     0.00

PARALLEL TASKS................59     0.01   FALLBACK - NO ENCLAVE............0
                                            FALLBACK - RLF LIMITED...........0
                        - - - - SYSPLEX - - - -
COORDINATOR TASKS.............0     0.00   FALLBACK - NO COORDN.............0
ASSISTING TASKS...............40     0.01   FALLBACK - ISOL RR/RS............0
MAXIMUM MEMBERS...............1     0.00   REDUCED  - VP SHORTAGE...........0
INTENDED PRLL GROUPS..........11     0.00
QUERY REFORMULATED (ENV).......0     0.00
QUERY REFORMULATED (BP)........0     0.00
```

Figure 34. TSTAT Parallelism (PRLL) Section

See “Parallelism” on page 154.

```
- - - - - - - - - - - - - - -STORED PROCEDURES- - - - - - - - - - - - - - - - -
                          TOTAL   AVERAGE                                TOTAL
SPAS CPU TIME.........1,190 ms    595 ms   SQL CALLS.......................2
SPAS IN-DB2 CPU TIME....879 ms    440 ms
SPAS WAITING ON TCB.......0 us      0 us   WAITS ON TCB....................0
                                           CALLS ABENDED...................0
                                           CALLS REJECTED..................0
                                           CALLS TIMED OUT.................0
```

Figure 35. TSTAT Routines (RTN) Section

See “Routines” on page 157.

```
- - - - - - - - - - - - - - - - DDF SUMMARY - - - - - - - - - - - - - - - - -
CURRENT LOCATION: DB2F              (ALLIED DIST REQUESTER)
                          TOTALS  AVERAGES                        FROM       TO
REMOTE LOCATION: DB1                          SQL STATEMENTS       N/A        2
REMOTE PROD-ID:  DSN03010                     ROWS                   0      N/A
DB AGENT CPU..........   32 ms     32 ms      BYTES              1,658    1,345
ELAPSED LOCAL.........00:03:53  00:03:53     TRANSACTIONS         N/A        1
ELAPSED REMOTE........00:03:53  00:03:53     MESSAGES               1        2
CONVERSATIONS QUEUED.........0      0.00      CONVERSATIONS        N/A        1
BLOCK FETCH%................0                COMMITS              N/A        0
MAX CONVERSATIONS...........1      1.00       ABORTS               N/A        0
UNSUCCESSFUL CONV'S.........0      0.00       BLOCK FETCH            0      N/A

TWO PHASE COMMIT (2PH):                       2PH PREPARE            0        0
                                              2PH LAST AGENT         0        0
REMOTE COORDINATOR                            2PH COMMITS            0        0
------------------                            2PH BACKOUTS           0        0
-- INDOUBT........           0                2PH FORGET RSP         0        0
-- COMMIT.........           0                2PH REQ COMMIT         0        0
-- ROLLBACK.......           0                2PH BACKOUT RSP        0        0
```

Figure 36. TSTAT DDF Summary (DDF) Section

See “DDF Summary” on page 158.

```
- - - - - - - - - - - DATABASE SUMMARY (DETAIL TRACE ONLY) - - - - - - - - - -   -
                          ----- INDX -----  ------ DATA -----  ------ WORK+ ----
DATABASE  PAGESET OBID  #SCAN #PAGE #ROWS  #SCAN #PAGE #ROWS  #SCAN #PAGE #ROWS
-------- -------- ----- ----- ----- -----  ----- ----- -----  ----- ----- -----
DSNDB06  SYSPLAN  ALL      21    68   485     20    13    14      0     0     0
DSNDB01  SCT02    ALL      18    36    18     15     3     9      0     0     0
DSNDB06  SYSDBASE ALL      13    28    13      7   109  2125      0     0     0
DSNDB07  DSN4K01    3       0     0     0      0     0     0      2     0    33
**** TOTALS *****          52   132   516     42   125  2148      2     0    33
```

Figure 37. TSTAT Database Summary Section

See “Database Summary” on page 161.

```
- - - - - - - - - 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
--------  -------- --- ----- ------ ---- -----  ------- -------- ---- -----
DSNDB01  DSNSCT02   0   0      0      1     0    19 ms    19 ms    0    0
DSNDB01  SCT02      2   0      0      1     0    24 ms    24 ms    0    0
**** TOTALS *****   2   0      0      2     0    43 ms    22 ms    0    0
```

Figure 38. TSTAT Database Lock and I/O Summary Section

See “Database Lock and I/O Summary” on page 162.

```
- - - - - - - - - - -  SORT SUMMARY (DETAIL TRACE ONLY) - - - - - - - - - - -
                                        SORT       SORT     RECORD
                                      ELAPSED    RECORDS     SIZE
                                      --------   --------   -------
NUMBER OF SORTS........1    AVERAGE   2,753 ms        33        84
AVG # WORK FILES.....2.0    MAXIMUM   2,753 ms        33        84
                            MINIMUM   2,753 ms        33        84
                            TOTAL     2,753 ms        33       N/A
```

Figure 39. TSTAT Sort Summary Section

See “Sort Summary” on page 164.

Select Code:

TSTAT

Parameter:

The following parameters can be specified:

trace id

Unique ID identifying the trace.

The TSTAT trace ID must match the trace ID of the ATRAC request. A blank can be used if the ATRAC ID is also blank.

This parameter is positional and a comma delimits the next parameter. For example, if there is no identifier, a comma must precede any other parameter. However, if there is an ID, it must be in the first position.

,DTL

Selects the data for the Database Summary, the Database Lock and I/O Summary, and the Sort Summary sections when viewing a history trace.

CAUTION:
This parameter requires a substantial amount of additional I/O.

,A|AUTHID=xxxxxxxx

Selects a subset of the trace entries by AUTHID. Specify one AUTHID, or a generic group of AUTHIDs by replacing character positions with a + (one position) or an * (multiple positions).

,C|CONNECT=xxxxxxxx
: Selects a subset of the trace entries by connection name. Specify one connect type (TSO, BATCH, DB2CALL, imsid, cicsjobname), or a generic group of connections by replacing character positions with a + (one position) or an * (multiple positions).

,P|PLAN=xxxxxxxx
: Selects a subset of the trace entries by plan. Specify one plan, or a generic group of plans by replacing character positions with a + (one position) or an * (multiple positions).

,L|LOC=xxxxxxxx
: Selects a subset of the trace entries by location. Specify one location, or a generic group of locations by replacing character positions with a + (one position) or an * (multiple positions).

,R|CORRID=xxxxxxxxxxxx
: Selects a subset of the trace entries by correlation ID. Specify one correlation ID, or a generic group of correlation IDs by replacing character positions with a + (one position) or an * (multiple positions).

,T|TIME=hhmm[-hhmm]
: Selects a subset of the trace entries by a start time or time period; hhmm indicates hours and minutes.

  Midnight wrap is supported where the start time is higher than the end time; for example, TIME=2000-3000. The end timestamp of an accounting record is used to assign it a time period.

  Note: If you request TSTAT for a trace log data set without specifying either a time qualification or the DTL parameter, only the summary statistics of the display are produced. This enables you to control the amount of resources used to produce the TSTAT display.

  The entire TSTAT display is produced when

  - A time qualification or the DTL parameter is specified for an historical trace log data set
  - The display is requested for an active trace, regardless of time qualification

,D|DAY=nn[-nn]
: Selects a subset of the trace entries by a day or range of days; nn can be a relative number from the date of the first accounting record in the trace buffer (first date = 1, up to a maximum of 99). If a time period is selected without a DAY specification, it defaults to the day of the most current record in the buffer.

  If there is data for only one day in the trace buffer, the format of the timestamp is hh:mm:ss. If there is data for more than one day, the format is dd-hh:mm, where dd is the relative day number from the date of the first accounting record in the trace buffer.

  Each accounting record traced is assigned to a time period based on its end timestamp. Therefore, the processing done for that unit of work is assigned to one interval, but it may have partially occurred during one or more previous time intervals depending on its elapsed time.

Note: Each keyword can be specified only once. If more than one selection keyword is specified, the result is ANDed.

Description:

The TSTAT display shows overall statistics and status of the trace.

The SCROLL field contains CSR and additional data can be seen by scrolling down with PF8/20. N/A in the SCROLL field indicates a summary trace.

The summaries are displayed in the following sequence:

- Base Section:
  - Trace status (SUMMARY STATISTICS - ALL TRACE ENTRIES)
  - Key indicators of performance problems
- Accounting Sections:
  - Elapsed time data if accounting class 2 or 3 is active (ELAPSED TIME ANALYSIS)
  - Statistics for the individual types of SQL statements executed (SQL STATEMENT EXECUTION COUNTS)
  - Buffer pool usage (BUFFER POOL ACTIVITY)
  - Lock usage (LOCK ACTIVITY)
  - Parallel CPU and I/O activity (PARALLELISM)
  - Routine statistics if stored procedure calls have been issued (ROUTINES)
  - DDF statistics if there is DDF activity for the displayed thread (DDF SUMMARY)
- Detail Trace Summary Sections:
  - Database summary for all records (DATABASE SUMMARY)
  - Synchronous reads and writes and lock suspensions (DATABASE LOCK AND I/O SUMMARY)
  - Summary of number of sorts, records sorted, sort elapsed time, and record size (SORT SUMMARY)

Note: The headings for detail trace summaries are shown with no data if the display is for a summary trace or if the ,DTL parameter is not specified for a detail trace.

Service Message(s):

A message in the parameter field shows the sequence number of the first row of statistics displayed and the total number of rows that can be displayed by this scrollable service.

Expand:

The TSTAT display can be EXPANDed to the following displays when the trace is being logged:

MON(WKLD)
: Active Timer Requests display of all active monitors in the DB2 workload (WKLD) area.

HISTORY
: Show the data from the current trace log data set for this trace (if you are viewing a current trace with logging, this provides access to data no longer in the online buffer). It is only shown when logging is active.

CURRENT
: The active trace data still in the buffers (if you are viewing a trace log data set for a still-active trace or a complete trace that has not yet been purged).

ACCOUNTING:
: The following sections format data from the DB2 accounting record.

ELAPSED
: TSTAT Elapsed Time Analysis display section (accounting classes 2 and 3 only)

SQLCOUNTS
: TSTAT SQL Statement Execution Counts display section

BPOOL
: TSTAT Buffer Pool Usage Analysis display section

LOCKS
: TSTAT Lock Activity display section

PRLL
: TSTAT Parallelism display section

RTN
: TSTAT Routines display section (only if stored procedure calls have been issued)

DDF
: TSTAT DDF Summary display section (available only if there is DDF activity for the displayed thread)

SUMMARIES:
: The following sections format data summarized from detail trace events and are available only if these events are traced.

SCANS
: TSTAT Database Summary display section

IO/LOCK
: TSTAT Lock and I/O Summary display section

SORTS
: TSTAT Sort Summary display section

I/O-DB/TS
: Display of I/O analysis by database and table space (DBIO) for the specified trace. (Available for an I/O detail trace only.)

Summary Statistics (Base Section)

The following areas are displayed on the initial screen where the most critical data is summarized.

Area 1 - Summary Statistics - All Trace Entries

This area displays the trace status and selected statistics summaries.

```
- - - - - - - - - - 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
-- IFI 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%
```

where

TERMINATIONS
The following fields provide statistics on the termination of the transactions in the trace:

FIRST END
Date and time the oldest entry in this trace ended.

TRACE END
Date and time the newest entry in this trace ended.

NUMBER TRANS
Total transactions traced for this trace.

COMMIT/ROLLBK
Total number of commits and total number of rollbacks for this trace.

NORMAL TERM
Number of normal transaction terminations. This total is divided into the following categories according the reason for termination:

NEW USER
Number of terminations due to the sign-on of a new user. This includes both sign-ons where the authorization ID is changed and those with the same authorization ID.

DEALLOC
Number of terminations due to deallocation. In this case, the application informed DB2 that it completed its work with DB2.

APPL END
Number of terminations due to end of task. In this case, the application program terminated without using system-directed access to end its connection to DB2 and MVS informed DB2 of the termination. The agent did not abend so it is considered a normal termination.

RESIGNON
Number of terminations due to partial sign-ons where the authorization ID remains the same (also called resign-ons).

DBAT INACT
Number of terminations due to a DDF thread becoming inactive.

IFI READ
Number of terminations due to Instrumentation Facility Interface (IFI) read requests.

ABNORMAL TERM
Number of abnormal transaction terminations.

IN DOUBT TERM
Number of indoubt transaction terminations. This condition occurs when DB2 fails after it has finished its phase 1 commit processing and before it has started phase 2.

ACTIVITY
These fields provide total, average, maximum, and minimum statistics for all transactions in the trace:

ELAPSED
Accumulation of the elapsed times for all transactions.

ELP-DB2
Accumulation of the elapsed time spent while processing in DB2 code for all transactions (available only if accounting class 2 is active).

CPU
CPU time for the processing address space for all transactions.

For TSO, IMS, or batch connections, this is the sum of the TCB and SRB time in the address space.

For CICS, the value reported is the TCB time minus the thread start time for the thread subtask processing the transaction. This reports the relative cost for the DB2 activity for the transaction. Any CICS CPU time for the transaction is not included in this value. In the event of thread reuse, this value can be distorted, as it is a total time for the thread subtask, not per CICS transaction.

CPU-DB2
Accumulated CPU time spent while processing in DB2 code for all transactions (available only if accounting class 2 is active).

WAITS
Accumulation of time spent waiting while in DB2 code for all transactions. (This field is zero unless accounting level 3 is active.)

Note: It is possible under some circumstances that DB2 wait time can exceed total elapsed time. This is possible because DB2 adds all wait times for all processes. If the waits for concurrent processes overlap, this time is counted multiple times.

SQL
Total number of SQL statements executed for all transactions.

GETPAGES
Number of GETPAGE requests for all transactions. This field counts conditional and unconditional requests (both successful and unsuccessful.)

SYNC RDS
Number of synchronous read I/Os performed by all transactions.

PFCH PGS
Total number of pages read asynchronously by prefetch for all transactions. This value includes all pages read by list, sequential, and dynamic prefetch.

UPD/COMT
Number of SQL update type statements (INSERTs, UPDATEs, DELETEs) per commit statement for all transactions.

BFR HIT RATIOS:
Includes the hit ratios for both the virtual pool and the corresponding hiperpool for all transactions.

VIRTUAL POOL HIT RATIO
Measures buffer pool performance. A high hit ratio means that a high percentage of reads are satisfied from the buffer pool without having to access external storage.

Note: Heavy prefetch activity can drive this value close to 0.

A pure random hit ratio cannot be calculated since the getpage count in the DB2 accounting record includes both random and sequential activity.

The hit ratio is computed as follows:

```
((A–B) / A) x 100
```

where

A Is the number of GETPAGE requests

B Is the number of synchronous reads and asynchronous pages read

HIPERPOOL HIT RATIO
This value is a measure of performance of the hiperpool. A high hit ratio means that a high percentage of reads are satisfied from the hiperpool. A hit ratio of 100% indicates that MVS has not stolen any of the expanded storage allocated to the hiperpool.

The hit ratio is computed as follows:

```
((A–B) / A) x 100
```

where

A Is the total number of successful reads from the hiperpool

B Is the number of hiperpool read failures

Area 2 - Key Indicators

This area shows the most important key indicators of failures or degradation.

```
- - - - - - 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
```

The key indicators displayed in this area are selected in the following order of importance:

- These indicators show the cause of degradation and are highlighted:
  - **TIMEOUT (or DEADLOCK) = nnnnnn**

    Indicates the number of units of work that failed because either a timeout condition or deadlock condition was detected.

    TIMEOUT means that a unit of work was suspended for a length of time longer than permitted by installation options.

    DEADLOCK means that two units of work attempted to access the same resource and prevented each other from completing normally.

    Action: N/A

    Field: QTXATIM, QTXADEA
  - **STORED PROC. FAILED = nnnnnn**

    Indicates the number of times a stored procedure call failed. Possible reasons for failure are as follows:

    - The stored procedure call terminated abnormally.
    - The stored procedure call was rejected.
    - The stored procedure call was timed out while it was waiting to be scheduled.

    Action: N/A

    Field: QXCALLAB, QXCALLRJ, QXCALLTO
  - **RID FAILURE - STORAGE**

    Indicates the number of times RID list processing failed for the units of work in this trace because not enough storage was available.

    Action: Review the storage allocations specified when DB2 was installed.

    Field: QXNSMIAP

– **RID FAILURE - NUMBER OF RIDS**

Indicates the number of times RID list processing failed for the units of work in this trace because one of the internal limits was exceeded.

The internal limits include the physical limit of the number of RIDs a RID list can contain and internal thresholds for the retrieval and manipulation of RIDs.

Action: N/A

Field: QXMRMIAP

– **CLAIM FAILURE**

Indicates the number of times CLAIM processing failed for the units of work in this trace.

Action: N/A

Field: QTXACLUN

– **DRAIN FAILURE**

Indicates the number of times DRAIN processing failed for the units of work in this trace.

Action: N/A

Field: QTXADRUN

– **PARALLEL CPU FALLBACK - nnnnnn**

Indicates the number of times parallel CPU processing was disabled for the units of work in this trace.

Action: N/A

Field: QXDEGENC, QXRLFDPA

– **PARALLEL I/O FALLBACK - nnnnnn**

Indicates the number of times parallel I/O processing was turned off and sequential mode was used for the units of work in this trace.

Action: N/A

Field: QXDEGBUF, QXDEGESA, QXDEGCUR

– **PARALLEL I/O REDUCED DEGREE - nnnnnn**

Indicates the number of times Parallel I/O processing was reduced from the planned degree to a lesser degree because of insufficient storage or insufficient buffers in the buffer pool.

Action: If this number is not zero, consider increasing the size of the buffer pool or specifying a different buffer pool for the table space.

Field: QXREDGRP

- **INCREMENTAL BINDS = nnnnnn**

  Indicates the number of incremental binds performed by the units of work in this trace.

  This indicates that the plan/package had become invalid for some reason, such as an ALTER on an index or table.

  Action: Nothing needs to be done for this plan, but you may want to explicitly bind any plans whose tables/indexes are being altered. Incremental binds can have a significant impact in an online transaction system.

  Field: QXINCRB

- **TOTAL DDL = nnnnnn**

  Indicates the number of Data Definition Language (DDL) statements performed by the units of work in this trace.

  DDL statements, such as CREATE TABLE, can have a significant impact on system performance. To perform DDL operations, exclusive locks need to be obtained against the DB2 catalog.

  Action: In an online transaction system, avoid designing transactions that use DDL operations.

  Field: QXCRTAB, QXCRINX, and so on

- **BUFFER INCOMPLETE**

  Indicates at least one of the transactions spans multiple trace log data sets. Accounting statistics data is available for those transactions only in the last log, and detail trace data is incomplete in each log.

  However, accounting data in the final log is complete and represents the entire transaction. The detail trace data in each log can be added together for total results.

  Action: Allocate a larger trace log data set.

  Field: N/A

• The following indicators are informational in nature and are not highlighted:

- **GRANTS/REVOKES = nnnnnn**

  Indicates the number of GRANT and/or REVOKE SQL statements issued by the units of work in this trace.

  Action: N/A

  Field: QXGRANT, QXREVOK

- **LOCK TABLE = nnnnnn**

  Indicates the number of LOCK TABLE SQL statements issued by the units of work in this trace.

  Action: Use LOCK TABLE carefully. It means that this unit of work owns exclusive control of the table, allowing no concurrent access.

  Field: QXLOCK

– **LOCK ESCALATIONS = nnnnnn**

Indicates the number of times the total locks per table have exceeded the installation limit for the units of work in this trace. When this limit is exceeded, DB2 promotes the current page locks to a single table space lock of the same type.

Action: Lock escalation usually indicates that the application has encountered an exceptional condition. If this occurs frequently, the design of the application should be reviewed.

Field: QTXALES, QTXALEX

– **PARALLEL I/O CONDITIONAL GETPAGE FAILURE**

Indicates that a conditional GETPAGE request was not satisfied in a buffer pool. This can occur only for parallel I/O operations.

When this condition occurs, it indicates that a prefetch operation failed to retrieve a page before the application required it.

Action: N/A

Field: QBACNGT

– **SQL: SELECT=nnnnnn, FETCH=nnnnnn**

Summary of the number of SQL SELECT and FETCH statements issued by the units of work in this trace.

In a DDF environment, the number of FETCH statements may not truly reflect the number of FETCHes issued by the application due to internal DDF processing.

Action: N/A

Field: QXSELECT, QXFETCH

– **SQL: INS=nnnnnn, UPD=nnnnnn, DEL=nnnnnn**

Summary of the number of SQL INSERT, UPDATE, and/or DELETE SQL statements issued by the units of work in this trace.

Action: N/A

Field: QXINSRT, QXUPDTE, QXDELET

– **SQL: DYNAMIC(PREPARE)=nnnnnn**

Summary of the number of PREPARE SQL statements issued by the units of work in this trace. Each PREPARE statement is equivalent to one dynamic SQL statement.

In a DDF environment, the number of PREPARE statements may not truly reflect the number of dynamic SQL statements issued by the application due to internal DDF processing.

Action: N/A

Field: QXPREP

– **I/O RSP: SYNC= time, ASYNC= time**

Indicates the average I/O response times the units of work in this trace experienced for synchronous and asynchronous requests.

Action: N/A

Field: For synchronous requests:
QWACARNE, QWACAWTI

For asynchronous requests:
QWACARNR, QWACARNW, QWACAWTR, QWACAWTW

– **LOCK SUSPENSIONS = nnnnnn**

Number of times a unit of work was suspended due to a lock or latch contention with another unit of work.

This number should be small, ideally zero.

Action: N/A

Field: QTXASLOC, QTXASLAT, QTXASOTH

– **PARALLELISM, MAINTASKS = nnnnnnn, SUBTASKS = nnnnnnn**

Indicates that this trace contains units of work that are the primary, or originating, unit of work in a group supporting a query or utility using parallel tasks. MAINTASKS shows the total number of main tasks created to support queries or utilities using parallel tasks in this trace. SUBTASKS shows the total number of subtasks created to support queries or utilities using parallel subtasks in this trace.

Action: N/A

Field: QWACPACE, QWACPCNT

– **SYSPLEX PARALLELISM - COORDINATOR = nnnnnn**

Indicates that queries are being processed across more than one DB2, where this DB2 was the coordinator (DB2 version 5 and later only).

Action: N/A

Field: QWDA

– **SYSPLEX PARALLELISM - ASSISTANT = nnnnnn**

Indicates that queries are being processed across more than one DB2, where this DB2 was an assistant (DB2 version 5 and later only).

Action: N/A

Field: QWDA

– **NUMBER OF DISTRIBUTED LOCATIONS = nnnnnn**

Number of separate DDF locations accessed by the units of work in this trace.

Action: N/A

Field: N/A

– **RID LIST PROCESSING USED = nnnnnn**

Number of times RID list processing was used for the units of work in this trace.

During RID list processing, DB2 produces a list of candidate record IDs from an index. The resulting RID list can be used to efficiently retrieve the qualifying rows.

Action: N/A

Field: QXMIAP

Accounting Sections

The following sections format data from the DB2 accounting record.

Elapsed Time Analysis

The Elapsed Time Analysis section shown in Figure 40 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the ELAPSED expand button

The data is displayed only if accounting class 2 or 3 is active. If accounting class 2 or 3 is not active, only the header is displayed.

```
- - - - - - - - ELAPSED TIME ANALYSIS (ACCTG CLASSES 2,3 ONLY)- - - - - - - - -

CATEGORY           #EVENTS AVG/EVENT  ELAPSED  %TOTAL
----------------   ------- --------- -------- ------- 0 ...25...50...75..100%
ELAPSED TIME                                           |                      |
  IN DB2                12     20 s  00:03:58   38.07 | *******              |
  IN APPLICATION        12     32 s  00:06:27   61.92 | ************         |
 --TOTALS--             12     52 s  00:10:24  100.00 | ******************** |
WAITS IN DB2 (LOCAL)                                   |                      |
  LOCK/LATCH            11  3,148 ms    35 s     5.54 | *                    |
  I/O WAIT             160     17 ms 2,652 ms    0.42 | <                    |
  LOG WRITE I/O          2  5,151 us    10 ms    0.00 |                      |
  OTHER READ            15     42 ms   632 ms    0.10 | <                    |
  OTHER WRITE            0      0 us     0 us    0.00 |                      |
  UNIT SWITCH EVENTS                                   |                      |
  ..COMMIT/ROLLBK       34     72 ms 2,438 ms    0.39 | <                    |
  ..OPEN/CLOSE          90  1,632 ms 00:02:27   23.53 | ****                 |
  ..SYSLGRNG            35     38 ms 1,347 ms    0.21 | <                    |
  ..DATASPACE MGR       89    467 ms    42 s     6.65 | *                    |
  ..OTHER               37     26 ms   950 ms    0.15 | <                    |
  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--       220  1,050 ms 00:03:51   37.02 | *******              |
*NOT ACCOUNTED                        1,464 ms    0.23 | <                    |
********************************* END OF DATA *********************************
```

Figure 40. TSTAT Elapsed Time Analysis (ELAPSED) Section

This area displays the elapsed time, number of events, average elapsed time per event, and percentage of total elapsed time for each of the following categories. The percentage of total elapsed time is also displayed graphically for each of these categories.

ELAPSED TIME

This information is available if accounting class 2 is active:

IN DB2

Elapsed time spent in DB2 processing (QWACASC).

In-DB2 time is elapsed time while performing DB2 work. The DB2 time is a percentage of the total elapsed time.

Note: DB2 combines elapsed times of related processes into a single in-DB2 elapsed time value; thus the in-DB2 elapsed time may exceed the total elapsed time. When this situation occurs, the percent of total elapsed time is set to 101 percent.

IN APPLICATION

Total elapsed time minus the time spent in DB2 (QWACESC - QWACBSC - QWACASC).

Application time is elapsed time spent in the application while NOT performing DB2 work.

Note: DB2 combines elapsed times of related processes into a single in-DB2 elapsed time value; thus the in-DB2 elapsed time may exceed the total elapsed time. When this situation occurs, the in-application time is not calculated and is displayed as asterisks.

TOTALS

Total elapsed time from the first connect to DB2 to thread termination (QWACESC - QWACBSC).

The time a thread was inactive waiting for work is included in this value.

This value is not accurate for threads that do not terminate, such as a CICS conversational transaction or an IMS WFI BMP.

Note: The percent of total elapsed time is always 100 percent. This measure is included to establish the scale.

WAITS IN DB2 (LOCAL)

The elapsed time for the following DB2 wait events is available only if accounting class 3 is active, as follows:

LOCK/LATCH

Elapsed time spent in DB2 waiting for a lock or latch (QWACAWTL).

This value indicates the elapsed time the thread waited for transaction locks and latches.

This time reflects the amount of time the thread was suspended because of an incompatible lock or latch mode.

I/O WAIT

Elapsed time spent in DB2 waiting for synchronous I/O (QWACAWTI).

Most of the time included in this value is synchronous read time, but under stress conditions it can reflect synchronous write time (when immediate write threshold is reached). During heavy update activity, it also can reflect wait for synchronous log I/O.

Note: Log waits are not included in this value for DB2 6.1 and later.

OTHER READ I/O

Elapsed time spent in DB2 waiting for read I/O, for I/O done by another thread such as sequential prefetch, sequential detection, or list prefetch (QWACAWTR).

This value is the time spent waiting for reads, which are being done under another thread, to complete. These waits may be for sequential prefetch, sequential detection, list prefetch, or for synchronous I/O done while executing a sequential prefetch operation.

Tuning Tip: Time reflected here does not reflect the duration of the I/O but simply the time the thread waited for completion of the I/O activity. If this number is high, parallel processing may assist in reducing the time.

OTHER WRITE I/O

Elapsed time spent in DB2 waiting for write I/O done by another thread (QWACAWTW).

Tuning Tip: Time reflected here should be small, since normal checkpoint and deferred write thresholds protect threads from waiting for asynchronous write time. If this number is large, the buffer pool size or deferred write threshold may need to be adjusted.

UNIT SWITCH EVENTS

In DB2 6.1 and later, the elapsed times spent in DB2 waiting for a synchronous execution unit switch to each of the following DB2 service tasks, which are shown individually:

- Commit, abort, or deallocation processing
- Open/close data set or HSM recall
- SYSLGRNG recording
- Data space manager services (which include define, extend, delete, and reset data set, and VSAM catalog access)
- Other DB2 service tasks

In DB2 5.1 and earlier, these waits are combined into a single value.

ARCH.LOG(QIS)

Elapsed time spent in DB2 waiting for processing of archive log mode (quiesce) commands for the thread (QWACALOG or QWAXALOG in DB2 6.1 and later).

This process externalizes buffers and switches the active log data set. Time reflected here is the time the thread waited, not the time for the command to complete executing.

ARCH.READ(TAPE)

Elapsed time spent in DB2 waiting for archive reads (QWACAWAR or QWAXAWAR in DB2 6.1 and later).

Time spent waiting for an archive tape is either for the completion of the RECOVER utility or for an application program, which has not committed, to read the archive tape in order to complete backout.

Tuning Tip: If this time is large, the application program should commit more often. It may also make sense to increase the size and number of active logs.

DRAIN LOCK

Elapsed time spent in DB2 waiting to acquire drain locks (QWACAWDR or QWAXAWDR in DB2 6.1 and later).

Processes, such as utilities that serialize events to page sets, must acquire a drain lock. This is the time spent waiting for this event.

Tuning Tip: To prevent delays, avoid running utilities during the times that heavy query activity is taking place.

CLAIM RELEASE

Elapsed time spent in DB2 waiting for claim locks to be released after a drain lock was requested (QWACAWCL or QWAXAWCL in DB2 6.1 and later).

Drainers must wait until the SQL claim count reaches zero. This is the time spent waiting for them to clear.

Tuning Tip: To prevent delays, avoid running utilities during the times that heavy query activity is taking place.

PAGELATCH CONT.

Elapsed time spent in DB2 because of page latch contention (QWACAWTP).

Accounting class 3 must be activated to obtain this time.

Tuning Tip: Page latching can be avoided by scheduling applications when there is a lower likelihood of latch contention.

SPAS SERVER TCB

Total elapsed time spent waiting for an available TCB before the Stored Procedure could be scheduled (QWACCAST).

This is the time spent waiting for an available TCB in the Stored Procedures Address Space to become available.

This time should be minimal depending on the workload to the Stored Procedures Address space.

WAITS IN DB2 (GLOBAL)

The elapsed time for the following global DB2 wait events is available only if accounting class 3 is active:

LOCK

Accumulated elapsed wait time caused by an IRLM lock suspension due to global lock contention which requires intersystem communication to resolve (QWACAWTJ).

This time is collected only when data sharing is enabled.

MESSAGE PROC.
Accumulated elapsed wait time caused by suspension for sending messages to other members in the data sharing group (QWACAWTG).

When tables in a data sharing group are CREATed, ALTERed, or DROPped in a data sharing group, intersystem messages must be sent to update the database descriptors (DBDs) in the other members' EDM pools. This is the time spent waiting for this message traffic to occur. This time is collected only when data sharing is enabled.

TOTAL WAITS
Total elapsed time for all DB2 wait events (local and global).

*NOT ACCOUNTED
Elapsed time that could not be accounted for by any DB2 measurements. This is usually some kind of MVS usage, such as DB2 being forced to wait on the MVS dispatcher.

This value is collected only if accounting classes 2 and 3 are activated and is calculated as follows:

In-DB2 Elapsed Time (QWACASC) - In-DB2 CPU (QWACAJST) - CLASS 3 WAIT

Tuning Tip: If this value is high, determine why MVS fails to dispatch DB2.

SQL Statement Execution Counts

The SQL Statement Execution Counts shown in Figure 41 section can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the SQLCOUNTS expand button

This section maintains counts for the individual types of SQL statements executed by all transactions in this trace. Both total and average counts are shown for DML statements.

```
- - - - - - - - - - - - SQL STATEMENT EXECUTION COUNTS - - - - - - - - - - - -
             TOTAL  AVERAGE        TOTAL   TOTAL
SELECT.........209      1.0   LOCK TABLE.............0   CREATE............4
INSERT.........155      0.8   GRANT/REVOKE............7   DROP..............1
UPDATE.........219      1.1   SET CURR.SQLID..........7   ALTER.............0
DELETE..........87      0.4   SET HOST VAR. .........12
                              SET CURR.DEGREE.........0   COMMENT ON.........0
PREPARE........247      1.2   SET CURR. RULES.........0   LABEL ON...........0
DESCRIBE........18      0.0   CONNECT TYPE 1..........0
                              CONNECT TYPE 2..........0   CREATE TEMP........0
OPEN...........352      1.8   SET CONNECTION..........0   RENAME TBL.........0
FETCH........2,248     11.7   RELEASE................0
CLOSE..........343      1.7   CALL...................0
ASSOCIATE LOCATOR.......0
ALLOCATE CURSOR.........0

** DML.......3,878     20.3   ** DCL................26   ** DDL............5
** REOPT.........0      0.0
                  - - - - DYNAMIC SQL CACHE TOTALS - - - -
STMT FOUND............0    KPDYN-PREP.AVOID.........0    STMT DISCRD........0
STMT NOT FOUND........0    KEEPDYN_IMPL.PREP........0    STMT PURGED........0
******************************** END OF DATA ********************************
```

Figure 41. TSTAT SQL Statement Execution Counts (SQLCOUNTS) Section

where

SELECT
: Number of SELECT statements executed.

INSERT
: Number of INSERT statements executed.

UPDATE
: Number of UPDATE statements executed.

DELETE
: Number of DELETE statements executed.

PREPARE
: Number of PREPARE statements executed.

DESCRIBE
: Number of DESCRIBE statements executed.

OPEN
: Number of OPEN statements executed.

FETCH
Number of FETCH statements executed.

CLOSE
Number of CLOSE statements executed.

DML
The sum of all Data Manipulation Language (DML) statements executed.

REOPTIMIZED
Indicates the total number of times reoptimization occurred for a query (DB2 Version 5 and later).

LOCK TABLE
Number of LOCK TABLE statements executed.

GRANT/REVOKE
Number of GRANT/REVOKE statements executed.

SET CURR.SQLID
Number of SET CURRENT SQLID statements executed.

SET HOST VAR.
Number of SET HOST-VARIABLE statements executed. (This value does not include the contents of the special register.)

SET CURR.DEGREE
Number of SET CURRENT DEGREE statements executed.

SET CURR. RULES
Number of SET CURRENT RULES statements executed (DB2 Version 4 or above).

CONNECT TYPE 1
Number of CONNECT TYPE 1 statements executed.

CONNECT TYPE 2
Number of CONNECT TYPE 2 statements executed.

SET CONNECTION
Number of SET CONNECTION statements executed.

RELEASE
Number of RELEASE statements executed.

CALL
Number of SQL CALL statements to a Stored Procedure executed (DB2 Version 4 or above).

ASSOCIATE LOCATOR
Number of ASSOCIATE LOCATOR statements executed (DB2 Version 5 and later).

ALLOCATE CURSOR
Number of ALLOCATE CURSOR statements executed (DB2 Version 5 and later).

DCL
Sum of all Data Control Language (DCL) statements executed.

CREATE
Number of CREATE statements executed.

DROP
Number of DROP statements executed.

ALTER
Number of ALTER statements executed.

COMMENT ON
Number of COMMENT ON statements executed.

LABEL ON
Number of LABEL ON statements executed.

CREATE TEMP
Number of CREATE GLOBAL TEMPORARY TABLE statements executed (DB2 Version 5 and later).

RENAME TBL
Number of RENAME TABLE statements executed (DB2 Version 5 and later).

DDL
Sum of all Data Definition Language (DDL) statements executed.

Dynamic SQL Cache Counts:

The following fields are available for DB2 Version 5 and later:

STMT FOUND
Indicates the number of times a prepare request was satisfied by making a copy from the prepared statement cache.

STMT NOT FOUND
Indicates the number of times a prepare request was received but a matching statement was not found in the prepared statement cache.

KPDYN-PREP.AVOID
Indicates the number of times that a prepare was avoided when these conditions existed:

- The KEEPDYNAMIC(YES) option was used along with prepare statement caching
- DB2 still had the application process copy of the executable version of the prepared statement

KEEPDYN_IMPL.PREP
Indicates the number of times that an implicit prepare was performed when these conditions existed:

- The KEEPDYNAMIC(YES) option was used
- An open, execute, or describe of a dynamic statement occurred after a commit
- DB2 no longer had a valid copy of the executable version of the prepared statement

STMT DISCRD
Indicates the number of times that a prepared statement was discarded due to the MAXKEEPD system limit being exceeded.

STMT PURGED
Indicates the number of times that a prepared statement was purged from the cache because a drop, alter, or revoke statement was issued on a dependent object.

Buffer Pool Activity

The Buffer Pool Activity section shown in Figure 42 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the BPOOL expand button

This section is repeated if there are more than two buffer pools with totals.

In this section, both total and average counts are shown for all transactions in the trace.

```
- - - - - - - - - - - - - - BUFFER POOL ACTIVITY- - - - - - - - - - - - - - - - -
                      ----- TOTAL -----  ------- BP0 -----  ------- BP2 -----
ACTIVITY                 TOTAL  AVERAGE     TOTAL  AVERAGE     TOTAL  AVERAGE
                      -------- --------  -------- --------  -------- --------
GETPAGES                11,767    147.0    11,630    145.3       137      1.7
SYNC READS                  84      1.0        81      1.0         3      0.0
GETPAGES/READIO          140.0              143.5               45.6

COND GP FAILURES             0      0.0         0      0.0         0      0.0
SEQ PREFETCH REQS        1,383     17.2     1,372     17.1        11      0.1
LIST PREFETCH REQS           0      0.0         0      0.0         0      0.0
DYNAMIC PREFETCHS            0      0.0         0      0.0         0      0.0
ASYNC PAGES READ        10,583    132.2    10,562    132.0        21      0.2
PAGES/PREFETCH REQ         7.6                7.6                1.9

PAGE UPDATES                 0      0.0         0      0.0         0      0.0
IMMEDIATE WRITES             0      0.0         0      0.0         0      0.0

HP SYNC READS              168      2.1       168      2.1         0      0.0
HP SYNC READ FAILS          10      0.1        10      0.1         0      0.0
HP ASYNC PAGES READ        250      3.1       250      3.1         0      0.0
HP WRITES                  249      3.1       249      3.1         0      0.0
HP WRITE FAILURES            0      0.0         0      0.0         0      0.0

            - - - GLOBAL BUFFER POOLS (DATA SHARING ONLY) - - -
ACTIVITY              ----- TOTAL -----  ------- BP0 -----  ------- BP1 -----
-------------------  -----------------  -----------------  -----------------
CF READS  (BUFFER CROSS INVALIDATION)
 -DATA RETURNED....         13      0.0        11      0.0         2      0.0
 -R/W INTEREST.....          6      0.0         6      0.0         0      0.0
CF READS  (DATA NOT IN BUFFER POOL)
 -DATA RETURNED....         62      0.3        62      0.3         0      0.0
 -R/W INTEREST.....        200      1.0       200      1.0         0      0.0
 -NO R/W INTEREST..          5      0.0         5      0.0         0      0.0
CF-WRITE (CHANGED)         160      0.8       127      0.6        18      0.0
CF-WRITE (CLEAN)             0      0.0         0      0.0         0      0.0
NBR UNREG PAGES...          50      0.2        50      0.2         0      0.0
NBR EXPLICIT XI...           0      0.0         0      0.0         0      0.0
NBR WRITES 2ND BP.           0      0.0         0      0.0         0      0.0
******************************** END OF DATA ********************************
```

Figure 42. TSTAT Buffer Pool (BPOOL) Section

where

GETPAGES
: Number of GETPAGE I/O requests.

SYNC READS
: Number of synchronous read I/Os.

GETPAGES/READIO
Number of GETPAGEs per synchronous READIO requests.

This ratio is only meaningful for random activity. If there is substantial asynchronous prefetch activity, review the value in ASYNC PAGES READ. You may want to calculate a total ratio including this value for a particular buffer pool. The VP HIT RATIO for the thread in the base section of the display includes both synchronous and asynchronous I/Os.

COND. GP FAILURES
Number of unsuccessful GETPAGE operations due to conditional GETPAGE requests.

SEQ. PREFETCH REQS
Number of sequential prefetch requests.

LIST PREFETCH REQS
Number of list prefetch requests.

DYNAMIC PREFETCHES
Number of dynamic prefetch requests.

ASYNC PAGES READ
Number of asynchronous pages read.

PAGES/PREFETCH REQ
The ratio of GETPAGEs per prefetch request.

PAGE UPDATES
Number of System Page Update requests done by this task. This counter is incremented by one each time a row in a page in the buffer pool is updated. It is also incremented for work file pages.

IMMEDIATE WRITES
Number of immediate (synchronous) write I/Os.

HP SYNC READS
Number of successful hiperpool synchronous reads.

HP SYNC READ FAILS
Number of unsuccessful hiperpool synchronous reads.

HP ASYNC PAGES READ
Number of hiperpool asynchronous pages read.

HP WRITES
Number of successful hiperpool writes.

HP WRITE FAILURES
Number of unsuccessful hiperpool writes.

Global Buffer Pool (Data Sharing Only):

The following fields are available if data sharing is active and has been used by this transaction.

CF READS (BUFFER CROSS INVALIDATION)
Coupling Facility READ requests required because the buffer was marked INVALID. The statistics are given for each of these circumstances:

-DATA RETURNED
Data is returned from the group buffer pool.

-R/W INTEREST
Data is not returned from the group buffer pool and a directory entry is created if it does not already exist, because another DB2 in the group has READ/WRITE interest in the page set or partition.

-NO R/W INTEREST
Data is not returned from the group buffer pool and no directory entry is created for this page, because no other DB2 in the group has READ/WRITE interest in the page set or partition.

CF READS (DATA NOT IN BUFFER POOL)
Coupling Facility READs necessary because the requested page was not found in the buffer pool. The statistics are given for each of these circumstances:

-DATA RETURNED
Data is returned from the Coupling Facility.

-R/W INTEREST
Data is not returned from the Coupling Facility and a directory entry is created if it does not already exist, because another DB2 in the group has READ/WRITE interest in the page set or partition.

-NO R/W INTEREST
Data is not returned from the Coupling Facility and no directory entry is created for this page, because no other DB2 in the group has READ/WRITE interest in the page set or partition.

CF-WRITE (CHGD PGS)
Number of changed pages written to the group buffer pool.

CF-WRITE (CLEAN PG)
Number of clean pages written to the group buffer pool. (DB2 writes clean pages for page sets and partitions defined with GBPCACHE ALL.)

NBR UNREG PAGES
Number of coupling facility requests to unregister a page. (DB2 version 5 and later)

NBR EXPLICIT XI
Number of explicit cross-invalidations. (DB2 version 6 and later)

NBR WRITES 2ND BP
Number of coupling facility requests to write changed pages to the secondary group buffer pool for duplexing. (DB2 version 6 and later)

Lock Activity

The Lock Activity section shown in Figure 43 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the LOCKS expand button

In this section, both total and average counts are shown for all transactions in the trace.

```
- - - - - - - - - - - - - - LOCK ACTIVITY - - - - - - - - - - - - - - - - - - -
                      TOTAL  AVERAGE                            TOTAL  AVERAGE
TIMEOUT.................2      0.0   MAX PAGE/ROW LOCKS..........109      1.3
DEADLOCK................0      0.0
                                     LOCK REQUEST...............548      6.8
SUSPEND-LOCK............2      0.0   UNLOCK REQUEST.............434      5.4
SUSPEND-LATCH...........1      0.0   QUERY REQUEST................0      0.0
SUSPEND-OTHER...........0      0.0   CHANGE REQUEST...............47      0.5
                                     OTHER REQUEST................0      0.0
ESCALATION(SHR)..........0      0.0
ESCALATION(EXCL).........0      0.0

DRAIN REQUEST...........0      0.0   DRAIN FAILURE................0      0.0
CLAIM REQUEST..........132      1.6   CLAIM FAILURE................0      0.0
                    - - - - GLOBAL LOCK ACTIVITY - - - -
LOCK REQUEST...........10      0.1   XES LOCK REQUEST...........234      2.9
LOCK CHANGE REQST........0      0.0   XES CHANGE REQUEST..........10      0.1
UNLOCK REQUEST..........0      0.0   XES UNLOCK REQUEST.........162      2.0

IRLM GLBL SUSPEND........0      0.0   XES GLOBAL SUSPEND...........0      0.0
FALSE CONTENTION........2      0.0

INCOMPAT. RETAINED......0      0.0   NOTIFY MESSAGES SENT.........0      0.0
******************************** END OF DATA ********************************
```

Figure 43. TSTAT Locks (LOCKS) Section

where

TIMEOUT
: Timeout counts.

DEADLOCK
: Deadlock counts.

SUSPEND-LOCK
: Number of suspends because of lock conflicts.

SUSPEND-LATCH
: Number of suspends because of latch conflicts.

SUSPEND-OTHER
: Number of suspends because of other conflicts.

ESCALATION(SHR)
: Number of lock escalations to shared mode.

ESCALATION(EXCL)
: Number of lock escalations to exclusive mode.

DRAIN REQUEST
Number of drain requests

CLAIM REQUEST
Number of claim requests

MAX PAGE/ROW LOCKS
Maximum number of page or row locks held.

LOCK REQUEST
Lock request count.

UNLOCK REQUEST
Unlock request count.

QUERY REQUEST
Query request count.

CHANGE REQUEST
Change request count.

OTHER REQUEST
Other IRLM request count.

DRAIN FAILURE
Number of drain requests that were unsuccessful.

CLAIM FAILURE
Number of claim requests that were unsuccessful.

Global Lock Activity (Data sharing Only):

The following fields are available if the system is part of a data sharing group:

LOCK REQUEST
Number of LOCK requests for physical locks (P-locks).

LOCK CHANGE REQUEST
Number of CHANGE requests for physical locks (P-locks).

UNLOCK REQUEST
Number of UNLOCK requests for physical locks (P-locks).

IRLM GLOBAL SUSPEND
Number of suspensions because of IRLM global resource contention. (IRLM lock states were in conflict.)

FALSE CONTENTION
Number of suspensions caused by false contentions.

This occurs when different resource names hash to the same entry in the Coupling Facility lock table. This causes MVS XES to detect contention on the hash class; however, when MVS XES determines that there is no real conflict on the resource, the contention is called *false*.

INCOMPATIBLE RETAINED
Number of global lock or change requests denied because of an incompatible retained lock.

XES LOCK REQUEST
Number of LOCK requests (both L-locks and P-locks) propagated to MVS XES synchronously (under the user's execution unit).

XES CHANGE REQUEST
Number of CHANGE requests (both L-locks and P-locks) propagated to MVS XES synchronously (under the user's execution unit).

XES UNLOCK REQUEST
Number of resources propagated to MVS XES synchronously (under the user's execution unit) from UNLOCK requests (both L-locks and P-locks).

XES GLOBAL SUSPEND
Number of suspensions because of MVS XES global resource contention. (MVS XES lock states were in conflict but IRLM lock states were not.)

NOTIFY MESSAGES SENT
Number of NOTIFY messages sent to other members of this data sharing group.

Parallelism

The Parallelism section shown in Figure 44 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the PRLL expand button

Note: With DB2 3.1, the section header reads: I/O PARALLELISM.

In this section, both total and average counts are shown for all transactions in the trace.

```
- - - - - - - - - - - - - - - - PARALLELISM - - - - - - - - - - - - - - - - - - -
                                TOTAL  AVERAGE                                TOTAL
MAXIMUM DEGREE (MAX=3)  ......42     0.01   FALLBACK - NO BUFFER............0
GROUPS EXECUTED...............21     0.00   FALLBACK - NO ESA SORT..........0
-- PLANNED DEGREE.............21     0.00   FALLBACK - AMBIG. CURSOR........0
-- REDUCED DEGREE..............0     0.00

PARALLEL TASKS................59     0.01   FALLBACK - NO ENCLAVE...........0
                                            FALLBACK - RLF LIMITED..........0
                           - - - - SYSPLEX - - - -
COORDINATOR TASKS.............0     0.00   FALLBACK - NO COORDN............0
ASSISTING TASKS...............40     0.01   FALLBACK - ISOL RR/RS...........0
MAXIMUM MEMBERS...............1     0.00   REDUCED  - VP SHORTAGE..........0
INTENDED PRLL GROUPS..........11     0.00
QUERY REFORMULATED (ENV).......0     0.00
QUERY REFORMULATED (BP)........0     0.00
```

Figure 44. TSTAT Parallelism (PRLL) Section

where

MAXIMUM DEGREE (MAX=n)
: Total degree of parallelism of all of the parallel groups found in this trace.

  MAX=n, where n is the maximum degree of parallelism encountered in this trace.

  These statistics indicate the extent to which queries were processed in parallel.

GROUPS EXECUTED
: Total number of parallel groups executed.

PLANNED DEGREE
: Total number of parallel groups that have a planned degree greater than one and were successfully executed at the planned degree.

REDUCED DEGREE
: Total number of parallel groups that have a planned degree greater than one but were processed to a parallel degree less than planned because of a storage shortage or contention on the buffer pool.

  If this field is not zero, consider increasing the size of the current buffer pool or reassigning the table space to another buffer pool.

FALLBACK - NO BUFFER
: Total number of parallel groups that have a planned degree greater than one but fell back to sequential mode because of storage shortage or contention on the buffer pool.

FALLBACK - NO ESA SORT
Total number of parallel groups that fell back to sequential mode due to lack of ESA sort support.

FALLBACK - AMBIG. CURSOR
Total number of parallel groups that fell back to sequential mode because the cursor is ambiguous (may be used for UPDATE or DELETE).

PARALLEL TASKS
Total number of subtasks that were created for the transactions in this trace to support parallel tasks for this query or utility. (Available only if accounting class 1 is active.)

FALLBACK - NO ENCLAVE
Total number of parallel groups executed in sequential mode because MVS/ESA enclave services were not available.

FALLBACK - RLF LIMITED
Indicates the number of transactions for which query parallelism was disabled by the Resource Limit Facility (RLF) for at least one dynamic select statement in the thread.

Sysplex Parallelism:

The following fields are available for DB2 version 5 and later:

COORDINATOR TASKS
Indicates the number of coordinator tasks that received a query for processing in the sysplex.

ASSISTING TASKS
Indicates the number of assisting tasks that processed a portion of a query in the sysplex.

MAXIMUM MEMBERS
Indicates the maximum number of DB2 members that participated in the processing of a query.

INTENDED PRLL GROUPS
Indicates the total number of parallel groups that were intended to run across the data sharing group. This value is only incremented by the coordinator member at run time.

FALLBACK - NO COORDN
Indicates the total number of parallel groups executed on a single DB2 (no sysplex).

FALLBACK - ISOL RR/RS
Indicates the total number of parallel groups executed on a single DB2 member due to repeatable-read or read-stability isolation (no sysplex).

REDUCED - VP SHORTAGE
Indicates the number of times that the parallelism coordinator had to bypass a DB2 when distributing tasks because there was not enough buffer pool storage on one or more DB2 members.

QUERY REFORMULATED (ENV)
Total number of parallel groups for which DB2 reformulated the parallel portion of the access path because the sysplex configuration at run time was different from the sysplex configuration at bind time.

This counter is incremented only by the parallelism coordinator at run time.

(DB2 6.1 and later)

QUERY REFORMULATED (BP)
Total number of parallel groups for which DB2 reformulated the parallel portion of the access path because there was not enough buffer pool resource.

This counter is incremented only by the parallelism coordinator at run time.

(DB2 6.1 and later)

Routines

The Routines section shown in Figure 45 can be viewed by any of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the RTN expand button

Note: This section is available only if the user connection has issued stored procedure calls.

```
- - - - - - - - - - - - - - -STORED PROCEDURES- - - - - - - - - - - - - - - - -
                            TOTAL   AVERAGE                                TOTAL
SPAS CPU TIME.........1,190 ms    595 ms   SQL CALLS........................2
SPAS IN-DB2 CPU TIME....879 ms    440 ms
SPAS WAITING ON TCB.......0 us      0 us   WAITS ON TCB.....................0
                                           CALLS ABENDED....................0
                                           CALLS REJECTED...................0
                                           CALLS TIMED OUT..................0
```

Figure 45. TSTAT Routines (RTN) Section

where

Stored Procedures:

SPAS CPU TIME
: Accumulated TCB time spent processing SQL CALL statements in the DB2 stored procedures address space (available only if accounting class 1 is active).

SPAS IN-DB2 TIME
: Accumulated TCB time spent in DB2 processing SQL statements issued by stored procedures (available only if accounting class 2 is active).

SPAS WAITING ON TCB
: Total elapsed time spent waiting for an available TCB before the stored procedure could be scheduled (available only if accounting class 3 is active).

SQL CALLS
: Number of SQL CALL statements executed.

CALLS ABENDED
: Number of times a stored procedure terminated abnormally.

CALLS REJECTED
: Number of times an SQL CALL statement was rejected because the procedure was in the ‘STOP ACTION(REJECT)’ state.

CALLS TIMED OUT
: Number of times an SQL CALL statement timed out while waiting to be scheduled.

WAITS ON TCB
: Number of times an SQL CALL statement had to wait for an available TCB before the stored procedure could be scheduled (available only if accounting class 3 is active).

DDF Summary

The DDF Summary section shown in Figure 46 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the DDF expand button

This section is available only if there is DDF activity for the displayed thread. It is repeated if there is more than one location.

In this section, both total and average counts are shown for all transactions in the trace.

Note: There can be multiple locations per thread and also multiple conversations per location.

```
- - - - - - - - - - - - - - - - DDF SUMMARY - - - - - - - - - - - - - - - - - -
CURRENT LOCATION: DB2F              (ALLIED DIST REQUESTER)
                        TOTALS  AVERAGES                        FROM       TO
REMOTE LOCATION: DB1F                       SQL STATEMENTS       N/A        2
REMOTE PROD-ID:  DSN03010                   ROWS                   0      N/A
DB AGENT CPU..........   32 ms     32 ms    BYTES              1,658    1,345
ELAPSED LOCAL.........00:03:53  00:03:53   TRANSACTIONS         N/A        1
ELAPSED REMOTE........00:03:53  00:03:53   MESSAGES               1        2
CONVERSATIONS QUEUED.........0      0.00   CONVERSATIONS        N/A        1
BLOCK FETCH%.................0             COMMITS              N/A        0
MAX CONVERSATIONS............1      1.00   ABORTS               N/A        0
UNSUCCESSFUL CONV'S..........0      0.00   BLOCK FETCH            0      N/A

TWO PHASE COMMIT (2PH):                     2PH PREPARE            0        0
                                            2PH LAST AGENT         0        0
REMOTE COORDINATOR                          2PH COMMITS            0        0
------------------                          2PH BACKOUTS           0        0
-- INDOUBT........           0              2PH FORGET RSP         0        0
-- COMMIT.........           0              2PH REQ COMMIT         0        0
-- ROLLBACK.......           0              2PH BACKOUT RSP        0        0
```

Figure 46. TSTAT DDF Summary (DDF) Section

where

CURRENT LOCATION:
Current 16-character location name.

REMOTE LOCATION:
Location name of the remote site with which the information is associated.

REMOTE PROD-ID:
Product ID of the remote location.

DB AGENT CPU
Database access agent CPU time at the remote site.

ELAPSED LOCAL
Elapsed time at the local site spent processing for the remote site.

ELAPSED REMOTE
Elapsed database access agent time at the remote site (accounting only). This data is not supported for application-directed access and is misleading if both system-directed access and application-directed access flows are used.

CONVERSATIONS QUEUED
Number of conversation requests queued by DDF waiting for allocation.

BLOCK FETCH%
Number of rows transmitted by block fetch as a percentage of the total rows transmitted.

MAX CONVERSATIONS
Maximum number of conversations open at any one time.

UNSUCCESSFUL CONV'S
Number of unsuccessful conversations.

SQL STATEMENTS
Number of SQL statements sent to and from remote site.

ROWS
Number of data rows sent to and from remote site.

BYTES
Number of bytes of data sent to and from remote site.

TRANSACTIONS
Number of transactions sent to and from remote site.

MESSAGES
Number of messages sent to and from remote site.

CONVERSATIONS
Number of conversations sent to and from remote site.

COMMITS
Number of commit requests sent to and from remote site.

ABORTS
Number of abort requests sent to and from remote site.

BLOCK FETCH
Number of blocks transmitted using block fetch.

TWO-PHASE COMMIT (2PH):

The following statistics are provided for two-phase commit operations:

REMOTE COORDINATOR

The following statistics are provided if there is a remote coordinator:

INDOUBT

Number of threads that went indoubt with the remote location as coordinator.

COMMIT

Number of commit requests sent to and from the remote site with the remote location as coordinator.

ROLLBACK

Number of rollback operations performed with the remote location as coordinator.

2PH PREPARE

Number of prepare requests sent to the participant and received from the coordinator for two-phase commit operations.

2PH LAST AGENT

Number of last agent requests sent to the coordinator and received from the initiator for two-phase commit operations.

2PH COMMITS

Number of commit requests sent to the participant and received from the coordinator for two-phase commit operations.

2PH BACKOUTS

Number of backout requests sent to the participant and received from the coordinator for two-phase commit operations.

2PH FORGET RSP

Number of forget responses sent to the coordinator and received from the participant for two-phase commit operations.

2PH REQ COMMIT

Number of request commit responses sent to the coordinator and received from the participant for two-phase commit operations.

2PH BACKOUT RSP

Number of backout responses sent to the coordinator and received from the participant for two-phase commit operations.

Detail Trace Summary Sections

The following sections format data summarized from detail trace events and are available only if these events are traced and the ,DTL parameter is specified.

Database Summary

The Database Summary section shown in Figure 47 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the SCANS expand button

This area displays a line of statistics for each database (DATABASE), page set (PAGESET), or data object identifier (OBID) accessed. There is a statistics column for each type of processing: index, data, or work (DSNDB07). The WORK+ column includes counts from temporary tables and transition tables, in addition to work files.

Note: This section is not shown for a summary trace. The heading line for this area appears with no data if the display is for a detail trace that does not include scan information or if the ,DTL parameter is not specified.

```
- - - - - - - - - - - DATABASE SUMMARY (DETAIL TRACE ONLY) - - - - - - - - - -   -
                        ----- INDX -----  ------ DATA -----  ------ WORK+ ----
DATABASE  PAGESET OBID  #SCAN #PAGE #ROWS  #SCAN #PAGE #ROWS  #SCAN #PAGE #ROWS
-------- -------- ----- ----- ----- -----  ----- ----- -----  ----- ----- -----
DSNDB06  SYSPLAN  ALL      21    68   485     20    13    14      0     0     0
DSNDB01  SCT02    ALL      18    36    18     15     3     9      0     0     0
DSNDB06  SYSDBASE ALL      13    28    13      7   109  2125      0     0     0
DSNDB07  DSN4K01    3       0     0     0      0     0     0      2     0    33
**** TOTALS *****          52   132   516     42   125  2148      2     0    33
```

Figure 47. TSTAT Database Summary Section

where

DATABASE
: Name of the database.

PAGESET
: Name of the page set.

OBID
: Five-digit data object identifier. The OBID can be matched with either the SYSIBM.SYSTABLES or SYSIBM.SYSINDEXES tables to translate the table or index name.

#SCAN
: Number of index, data, or work scans performed

#PAGE
: Number of index, data, or work pages accessed which includes pages scanned for referential integrity

#ROWS
: Number of rows processed which includes rows processed for referential integrity

Database Lock and I/O Summary

The Database Lock and I/O Summary section shown in Figure 48 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the IO/LOCK expand button

This area displays a row of statistics for each database and page set combination.

Note: This area is not shown for a summary trace. The heading line for this area appears with no data if the display is for a detail trace that does not include these events or if the ,DTL parameter is not specified.

```
- - - - - - - - - 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
--------  -------- --- ----- ------ ---- -----  ------- -------- ---- -----
DSNDB01  DSNSCT02   0   0      0      1     0    19 ms    19 ms    0    0
DSNDB01  SCT02      2   0      0      1     0    24 ms    24 ms    0    0
**** TOTALS *****   2   0      0      2     0    43 ms    22 ms    0    0
```

Figure 48. TSTAT Database Lock and I/O Summary Section

where

LOCKS
The following statistics are displayed for each database and page set combination:

Note: Zero values in the DATABASE or PAGESET field indicates the suspension or lock request is an internal latch to access a root index page.

MAX.
Maximum number of locks held

SUSP.
Number of lock suspensions

TM/OUT
Number of lock timeouts

SYNC. I/O
The following data is available only if data collection of I/O events was specified when the trace request was started. If no I/O events were collected, N/A is displayed in these fields.

READ
Number of synchronous READ I/Os

WRITE
Number of synchronous WRITE I/Os

ELAPSED
Total elapsed time for synchronous I/Os

AVG.ELAP
Average elapsed time per synchronous I/O request

ASYNC I/O

The following data is available only if data collection of I/O events was specified when the trace request was started. If no I/O events were collected, N/A is displayed in these fields.

READ

Number of asynchronous READ requests. These can be sequential prefetch, list prefetch, or dynamic sequential detection.

PAGES

Number of pages read asynchronously. Multiple pages can be read with one async READ request.

Sort Summary

The Sort Summary section shown in Figure 49 can be viewed by either of these methods:

- Scrolling down from the initial TSTAT display
- Selecting the SORTS expand button

This area displays statistics for internal sorts done by DB2.

Note: This area is not shown for a summary trace. The heading line for this area appears with no data if the display is for a detail trace that does not include these events or if the ,DTL parameter is not specified.

```
- - - - - - - - - - -  SORT SUMMARY (DETAIL TRACE ONLY) - - - - - - - - - - -
                                          SORT       SORT     RECORD
                                        ELAPSED    RECORDS     SIZE
                                        --------   --------   -------
NUMBER OF SORTS........1    AVERAGE   2,753 ms        33        84
AVG # WORK FILES.....2.0    MAXIMUM   2,753 ms        33        84
                            MINIMUM   2,753 ms        33        84
                            TOTAL     2,753 ms        33       N/A
```

Figure 49. TSTAT Sort Summary Section

where

NUMBER OF SORTS
: Total number of internal sorts done

AVG # WORK FILES
: Average number of sort work files used

The average, maximum, minimum, and total values are calculated for the following:

SORT ELAPSED
: Elapsed time for sort

SORT RECORDS
: Number of records sorted

RECORD SIZE
: Size of records sorted

Chapter 12. Viewing Workload History Summaries

To view workload history summaries, expand from LTRAC to one of the TSUMx displays.

TSUMx—Trace Summary Displays

If you need a global view of DB2 activity to do a detailed analysis of past workload-related problems, you can request online summary displays of workload history. These online workload history summaries are collected by summary or detail traces (by total system or selected workloads). Trace entries can be summarized by:

- Authorization ID (TSUMA service)
- Connect type (TSUMC service)
- Plan (TSUMP service)
- Location (TSUML service)
- Correlation ID (TSUMR service)
- Time (TSUMT service - selectable interval of minutes or hours)

These history summaries can be cursor-selected from the EXPAND line of the LTRAC display service. They can be displayed either in numerical format (see Figure 50) or in graphical format (see Figure 51 on page 168 and Figure 52 on page 169).

The requests can be ad hoc. You can use the trace summary services as you need them to analyze DB2 activity over time, by application, or by user instead of waiting for batch reports.

Display Fields

The columns for all trace summary displays include the workload indicators shown in Figure 50.

```
BMC Software --------------TRACE SUMMARY BY xxxxxx  -------------  RX AVAILABLE
SERV ==> TSUMx           INPUT    14:22:27  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> traceid,SORT=xx,GRAPH=NO                     ROW 1 OF 5 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY               ENTRIES IN DATASET 1 - 132
 19MAR01                      TRACE HISTORY    19MAR01 - 17MAR2001
           ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
 sumkey    COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
 -------- ------ -------- -------- ------ ------ -------- -------- ------ ------
 xxxx          2 5,516 ms   307 ms    3.0   70.0    11 s    615 ms      6    140
```

Figure 50. Trace Summary Display Fields

A trace summary display shows each summarized trace entry in a row. The entries are shown by the following:

summary key
: The summary key (represented by sumkey in Figure 50) can be:

AUTHID
: (TSUMA) authorization ID

CONNECT
(TSUMC) connection name

PLAN
(TSUMP) plan name

LOCATION
(TSUML) location name (first eight characters only)

CORR
(TSUMR) correlation ID (first eight characters only are displayed; however, all twelve characters are used and sorted on)

INTERVAL START
(TSUMT) interval start time

ENTRY COUNT
Number of trace entries summarized and sorted by summary key.

AVG ELAPSED
Average elapsed time as recorded in the traced DB2 accounting records sorted by summary key. It adjusts to seconds (s), milliseconds (ms), or microseconds (us) depending on the value. If the time exceeds 60 seconds, it is shown in the format hh:mm:ss.

AVG CPU
Average CPU time as recorded in the traced DB2 accounting records sorted by summary key. It adjusts to seconds (s), milliseconds (ms), or microseconds (us) depending on the value. CPU time includes all TCB and SRB time in all address spaces needed by DB2 to process the transaction.

AVG #STMTS
Average number of SQL statements issued by the transactions traced for this summary key.

AVG GETPGS
Average number of GETPAGE requests issued by the transactions traced for this summary key.

TOTAL ELAPSED
Total elapsed time as recorded in the traced DB2 accounting records for this summary key. It adjusts to seconds (s), milliseconds (ms), or microseconds (us) depending on the value. If the time exceeds 60 seconds, it is shown in the format hh:mm:ss.

TOTAL CPU
Total CPU time as recorded in the traced DB2 accounting records for this summary key. It adjusts to seconds (s), milliseconds (ms), or microseconds (us) depending on the value. CPU time includes all TCB and SRB time in all address spaces for processing the transaction, as seen by DB2.

TOTAL #STMTS
Total number of SQL statements issued by the transactions traced for this summary key.

TOTAL GETPGS
Total number of GETPAGE requests issued by the transactions traced for this summary key.

Service Messages

All trace summary services issue two service messages. One indicates the range of summary lines available for display and the other indicates the amount of entries in the trace buffer. These messages are displayed in front of the SCROLL field. The first message shows the sequence number of the first trace buffer entry displayed and the total number of entries that can be displayed. A message just below it shows the sequence number of the first and last entries currently available in the trace buffer.

Note: Sequence numbers are assigned internally to each trace entry when tracing starts. If the entries wrapped in the buffer (WRAP=YES) and overlaid earlier entries, the first sequence number in the buffer is no longer 1.

Scrolling

CSR in the SCROLL field indicates the display is scrollable. The display is refreshed whenever ENTER is pressed and no line is selected by cursor placement.

If the number of entries do not fit on one screen, the list can be scrolled as described in "Scrolling a Display" in *Using MAINVIEW.*

Using EXPAND

These services can be EXPANDed to the following displays:

LINESEL(LTRAC)
: Detailed display of information about a specific summary key (LTRAC display with only the trace entries summarized for the selected summary key).

  Position the cursor anywhere in a summary line for a summary key entry and press ENTER. PF3 (END) from that display returns to this list.

  For example, in Figure 53 on page 172, if you tab to the first line and press ENTER, all LTRAC entries with an authorization ID of CIR8 are shown.

HISTORY
: Show the data from the current trace log data set for this trace (if you are viewing a current trace with logging, this provides access to data no longer in the online buffer). It is only shown when logging is active.

CURRENT
: Active trace data still in the buffers (if you are viewing a trace log data set for a still-active trace or a complete trace that has not yet been purged).

Displaying Graphics

The default trace summary displays show numerical data for each of the key values from any trace. They display both average and total values. (See Figure 50 on page 165.)

To see this same data displayed graphically, use the GRAPH parameter:

```
,GRAPH=xxx
```

where xxx can be any of the following:

AVG
: Displays the Entry Count (number of threads) and all average values graphically.

TOT
: Displays the Entry Count (number of threads) and all total values graphically.

NO
: Default. Displays all values numerically.

Note: For the same trace request, the latest value of GRAPH you used is preserved as the default until you overtype it. Requesting Option 6, GRAPH, from the Primary Option Menu sets it to GRAPH=AVG.

Samples of the TSUMT display with average and total values presented graphically are shown in Figure 51 and Figure 52 on page 169.

- The values are plotted as percentages of the maximum value measured.
- The maximum is shown in the column header.
- Values over 90% of the maximum are red (or highlighted).
- Values over 70% of the maximum are yellow.
- Values less than 70% are green.
- Values less than 10% but non-zero show < in the first graph position.

GRAPH=AVG displays as follows:

```
BMC Software -------------- TRACE SUMMARY BY TIME   -----------PERFORMANCE MGMT
SERV ==> TSUMT           INPUT    15:38:38  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> THRDHIST,SORT=ISD,I=1H,GRAPH=AVG             ROW 1 OF 2 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY       ENTRIES IN BUFFER  1 - 53
19MAR01                        THREAD HISTORY         19MAR01 - 17MAR2001
INTERVAL     #THREADS    AVG ELAPSED      AVG CPU    AVG #STMTS    AVG GETPGS
START              38      00:05:12        464 ms          34.9          67.3
--------  |----.-----|  |----.-----|  |----.-----|  |----.-----|  |----.-----|
19:00:00   ********      <             *****         **            ******
18:00:00   ****          **            *             ***           *****
17:00:00   *             *             <             ******        ****
16:00:00   *******       <             <             *             **
15:00:00   ****          *             **            **********    ******
14:00:00   **********    **********    **********    **            **********
```

Figure 51. Sample Trace Summary Display with GRAPH=AVG

GRAPH=TOT displays as follows:

```
BMC Software -------------- TRACE SUMMARY BY TIME   -----------PERFORMANCE MGMT
SERV ==> TSUMT           INPUT    15:38:38  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> THRDHIST,SORT=IS,I=1H,GRAPH=TOT              ROW 1 OF 2 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY       ENTRIES IN BUFFER  1 - 53
19MAR01                        THREAD HISTORY         19MAR01 - 17MAR2001
INTERVAL     #THREADS    TOT ELAPSED      TOT CPU    TOT #STMTS    TOT GETPGS
START              38      03:17:59         17 s            616         2,560
--------  |----.-----|  |----.-----|  |----.-----|  |----.-----|  |----.-----|
14:00:00   **********    **********    **********    **********    **********
15:00:00   ****          *             **            *****         *****
16:00:00   *******       <             *****         **            **
17:00:00   *             *             ****          ****          ****
18:00:00   ****          **            ***           ******        *****
19:00:00   ********      <             *             *******       *******
```

Figure 52. Sample Trace Summary Display with GRAPH=TOT

Option 6, GRAPH, from the Primary Option Menu provides direct access to a graphical presentation of the Trace Summary by Time (TSUMT) display as shown in Figure 51 on page 168. It shows recent thread history summarized by 10-minute intervals, sorted in descending sequence. When accessed directly from the Primary Option Menu, this display has the following default parameters:

```
SERV ==> TSUMT
PARM ==> THRDHIST,SORT=ISD,INTVL=10M,GRAPH=AVG
```

Note: The default trace identifier requested is THRDHIST, a continuous summary trace defined in the sample block BLKDMRW in the BBPARM data set. This block is predefined in AutoCustomization for auto-start. However, you can change the default options of this trace in BLKDMRW if desired.

Sorting the Display

The trace summary displays show a list of scrollable, summarized trace entries sorted by summary key (first display column) for the selected trace summary (AUTHID for TSUMA, CONN for TSUMC, PLAN for TSUMP, LOCATION for TSUML, CORR for TSUMR, TIME for TSUMT). Any of the column data in the display can be sorted by typing the first two characters of the column heading with the SORT parameter as:

```
,SORT=cc
```

where cc can be any of the following two characters:

AU|CO|PL|LO|CR|IS
: Sorts the list alphabetically by authorization ID (AUTHID column in TSUMA service), connection name (CONNECT column in TSUMC service), plan name (PLAN column in TSUMP service), location name (LOCATION column in TSUML service), correlation ID (CORR column in TSUMR service), or by ascending time (INTERVAL START column in TSUMT service).

EC
: Sorts the list in a numerically descending order by entry count (ENTRY COUNT column).

AE
: Sorts the list in a numerically descending order by average elapsed time (AVG ELAPSED column).

AC
: Sorts the list in a numerically descending order by average CPU time (AVG CPU column).

A#
: Sorts the list in a numerically descending order by average number of statements (AVG #STMTS column).

AG
: Sorts the list in a numerically descending order by average number of GETPAGEs (AVG GETPGS column).

TE
: Sorts the list in a numerically descending order by total elapsed time (TOTAL ELAPSED column).

TC
: Sorts the list in a numerically descending order by total CPU time (TOTAL CPU column).

T#
: Sorts the list in a numerically descending order by total number of statements (TOTAL #STMTS column).

TG
: Sorts the list in a numerically descending order by total number of GETPAGEs (TOTAL GETPGS column).

ISD
: Interval Start Descending. Displays the most current measurements at the top (TSUMT only).

Note: You also can sort on each column by tabbing to the column header and pressing ENTER.

Selecting a Subset of Trace Entries

The following selection keywords can be typed directly in the parameter field:

,A|AUTHID=xxxxxxxx
: Selects a subset of the trace entries by authorization ID. Specify one authorization ID, or a generic group of authorization IDs by replacing character positions with a + (one position) or an * (multiple positions).

,C|CONNECT=xxxxxxxx
: Selects a subset of the trace entries by connect type. Specify one connect type (TSO, BATCH, DB2CALL, imsid, cicsjobname) or a generic group of connections by replacing character positions with a + (one position) or an * (multiple positions).

,P|PLAN=xxxxxxxx
: Selects a subset of the trace entries by plan. Specify one plan or a generic group of plans by replacing character positions with a + (one position) or an * (multiple positions).

,L|LOC=xxxxxxxx
: Selects a subset of the trace entries by location. Specify one location, or a generic group of locations by replacing character positions with a + (one position) or an * (multiple positions).

,R|CORRID=xxxxxxxxxxxx
: Selects a subset of the trace entries by correlation ID. Specify one correlation ID, or a generic group of correlation IDs by replacing character positions with a + (one position) or an * (multiple positions).

,T|TIME=hhmm[-hhmm]
: Selects a subset of the trace entries by a start time or time period, where hhmm indicates hours and minutes.

  Midnight wrap is supported, where the start time is higher than the end time; for example, TIME=2000-3000. The end timestamp of an accounting record is used to assign it a time period.

,D|DAY=nn[-nn]
: Selects a subset of the trace entries by a day or range of days, where nn can be a relative number from the date of the first accounting record in the trace buffer (first date = 1, up to a maximum of 99). If a time period is selected without a DAY specification, it defaults to the day of the most current record in the buffer.

Note: Each keyword can be specified only once. If more than one keyword is specified, the result is ANDed.

The parameter field is filled in automatically with any of these keywords when a series of summarizations and selections are made; for example:

- From LTRAC, select PLAN (TSUMP - Trace Summary by Plan) to see all trace entries.
- From TSUMP, sort by average CPU (SORT=AC) and cursor select (expand) the line for the plan with the highest average CPU. This returns to LTRAC but displays only the trace entries for that one plan. The parameter field is filled in with PLAN=xxxxxxxx.
- From this LTRAC subset, select AUTH (TSUMA - Trace Summary by AUTHID) to see a summary of all trace entries for this plan only. The parameter field is still filled in with PLAN=xxxxxxxx.

TSUMA—Trace Summary by Authorization ID (AUTHID)

The Trace Summary by AUTHID panel is a scrollable list of trace entries summarized and sorted by authorization ID (AUTHID column). Both totals and averages of key workload indicators are shown (see "Display Fields" on page 165).

```
BMC Software --------------TRACE SUMMARY BY AUTHID  -------------  RX AVAILABLE
SERV ==> TSUMA           INPUT    14:22:27  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SORT=AU,GRAPH=NO                      ROW 1 OF 5 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY                ENTRIES IN DATASET 1 - 132
 19MAR01                PBCRCP - SQL/SCAN/IO    19MAR01 - 17MAR01
           ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
 AUTHID    COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
 -------- ------ -------- -------- ------ ------ -------- -------- ------ ------
 CIR8          2 5,516 ms   307 ms    3.0   70.0    11 s    615 ms      6    140
 D8CS         22 6,380 ms    18 ms    7.5   10.0 00:02:20   398 ms    165    220
 NHJ1         26    54 s    214 ms   29.5  102.2 00:23:39 5,581 ms    768  2,659
 NHJ2          2 2,930 ms   430 ms  100.5  108.0 5,860 ms   860 ms    201    216
 WRD2          1 02:56:33    11 s     0.0    0.0 02:56:33    11 s       0      0
                         *** END OF SUMMARY ENTRIES ***
```

Figure 53. Trace Summary by Authorization ID

Select Code:

TSUMA

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace. A blank can be an identifier. This parameter is the first positional parameter; a comma delimits the next parameter. If there is no identifier, a comma must precede any other parameter.

,GRAPH=NO|AVG|TOT
: Displays a graphic presentation of the Entry Count (number of threads) followed by either the averages (AVG) or totals (TOT). NO requests a numerical display of both average and total values. Default is NO.

,SORT=AU|EC|AE|AC|A#|AG|TE|TC|T#|TG
: When the list of summarized trace entries is displayed initially, the list is sorted alphabetically by authorization ID (default) and ,SORT=AU is displayed in the PARM field (the comma is a required positional parameter for SORT). SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 169.

"Selecting a Subset of Trace Entries" on page 171 provides more information about the following parameters that applies to all the trace summary services:

,A|AUTHID=xxxxxxxx
: Selects the specified authorization ID only

,C|CONNECT=xxxxxxxx
: Selects the specified connection name only

,P|PLAN=xxxxxxxx
: Selects the specified plan name only

,L|LOC=xxxxxxxx
: Selects the specified location name only

,R|CORRID=xxxxxxxxxxxx
Selects the specified correlation ID only

,T|TIME=hhmm[-hhmm]
Selects the specified time only

,D|DAY=nn[-nn]
Selects the specified day only

Summary Key:
AUTHID is the first column of this display. The summary key used for the default sort of summarized trace entries is shown in the first column. The summary key for the TSUMA service is authorization ID.

Description:
Summarizes the traced transactions by authorization ID. The display is a scrollable list of summary lines for each unique authorization ID found in the trace entries.

TSUMC—Trace Summary by Connection Name (CONNECT)

The Trace Summary by Connect panel is a scrollable list of trace entries summarized and sorted by connection name (CONNECT column). Both totals and averages of key workload indicators are shown (see "Display Fields" on page 165).

```
BMC Software --------------TRACE SUMMARY BY CONNECT -------------  RX AVAILABLE
SERV ==> TSUMC           INPUT    14:24:47  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SORT=CO,GRAPH=NO                      ROW 1 OF 5 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY                ENTRIES IN DATASET 1 - 132
 19MAR01                PBCRCP - SQL/SCAN/IO    19MAR01 - 17MAR01
           ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
 CONNECT   COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
 -------- ------ -------- -------- ------ ------ -------- -------- ------ ------
 BATCH        23 00:01:00   205 ms   29.0  104.4 00:23:15 4,730 ms    667  2,403
 CICS3112     22 6,380 ms    18 ms    7.5   10.0 00:02:20   398 ms    165    220
 DB2CALL       1 02:56:33    11 s     0.0    0.0 02:56:33    11 s       0      0
 TSO           5 5,611 ms   390 ms   61.6   93.6    28 s  1,953 ms    308    468
 UTILITY       2 6,740 ms   186 ms    0.0   72.0    13 s    372 ms      0    144
                         *** END OF SUMMARY ENTRIES ***
```

Figure 54. Trace Summary by Connection Name

Select Code:

TSUMC

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace. A blank can be an identifier. This parameter is the first positional parameter; a comma delimits the next parameter. If there is no identifier, a comma must precede any other parameter.

,GRAPH=NO|AVG|TOT
: Displays a graphic presentation of the Entry Count (number of threads) followed by either the averages (AVG) or totals (TOT). NO requests a numerical display of both average and total values. Default is NO.

,SORT=CO|EC|AE|AC|A#|AG|TE|TC|T#|TG
: When the list of summarized trace entries is displayed initially, the list is sorted alphabetically by connection name (default) and ,SORT=CO is displayed in the PARM field (the comma is a required positional parameter for SORT). SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 169.

"Selecting a Subset of Trace Entries" on page 171 provides more information about the following parameters that applies to all the trace summary services:

,A|AUTHID=xxxxxxxx
: Selects the specified authorization ID only

,C|CONNECT=xxxxxxxx
: Selects the specified connection name only

,P|PLAN=xxxxxxxx
: Selects the specified plan name only

,L|LOC=xxxxxxxx
: Selects the specified location name only

,R|CORRID=xxxxxxxxxxxx
Selects the specified correlation ID only

,T|TIME=hhmm[-hhmm]
Selects the specified time only

,D|DAY=nn[-nn]
Selects the specified day only

Summary Key:
CONNECT is the first column of this display. The summary key used for the default sort of summarized trace entries is shown in the first column. The summary key for the TSUMC service is connection name (default) as:

- TSO
- BATCH
- DB2CALL (Call Attach Facility)
- IMS Subsystem ID
- CICS jobname

Description:
Summarizes the traced transactions by connection name. The display is a scrollable list of summary lines for each unique connection name found in the trace entries.

TSUML—Trace Summary by Location Name (LOCATION)

The Trace Summary by Location panel is a scrollable list of trace entries summarized and sorted by the first eight characters in the location name (LOCATION column). Both totals and averages of key workload indicators are shown (see "Display Fields" on page 165).

```
BMC Software --------------  TRACE SUMMARY BY LOC   -------------  RX AVAILABLE
SERV ==> TSUML           INPUT    14:25:20  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SORT=LO,GRAPH=NO                      ROW 1 OF 1 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY                ENTRIES IN DATASET 1 - 132
 19MAR01                PBCRCP - SQL/SCAN/IO    19MAR01 - 17MAR01
           ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
 LOCATION  COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
 -------- ------ -------- -------- ------ ------ -------- -------- ------ ------
 NEW_YORK     23 00:01:00   205 ms   29.0  104.4 00:23:15 4,730 ms    667  2,403
 DALLAS       22 6,380 ms    18 ms    7.5   10.0 00:02:20   398 ms    165    220
 CHICAGO       1 02:56:33    11 s     0.0    0.0 02:56:33    11 s       0      0
 LOS_ANGE+     5 5,611 ms   390 ms   61.6   93.6    28 s  1,953 ms    308    468
 SEATTLE       2 6,740 ms   186 ms    0.0   72.0    13 s    372 ms      0    144
                         *** END OF SUMMARY ENTRIES ***
```

Figure 55. Trace Summary by Location Name

Select Code:

TSUML

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace. A blank can be an identifier. This parameter is the first positional parameter; a comma delimits the next parameter. If there is no identifier, a comma must precede any other parameter.

,GRAPH=NO|AVG|TOT
: Displays a graphic presentation of the Entry Count (number of threads) followed by either the averages (AVG) or totals (TOT). NO requests a numerical display of both average and total values. Default is NO.

,SORT=LO|EC|AE|AC|A#|AG|TE|TC|T#|TG
: When the list of summarized trace entries is displayed initially, the list is sorted alphabetically by location name (default) and ,SORT=LO is displayed in the PARM field (the comma is a required positional parameter for SORT). SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 169.

"Selecting a Subset of Trace Entries" on page 171 provides more information about the following parameters that applies to all the trace summary services:

,A|AUTHID=xxxxxxxx
: Selects the specified authorization ID only

,C|CONNECT=xxxxxxxx
: Selects the specified connection name only

,P|PLAN=xxxxxxxx
: Selects the specified plan name only

,T|TIME=hhmm[-hhmm]
: Selects the specified time only

,D|DAY=nn[-nn]
Selects the specified day only

,L|LOC=xxxxxxxx
Selects the specified location name only

,R|CORRID=xxxxxxxxxxxx
Selects the specified correlation ID only

Summary Key:
LOCATION is the first column of this display. The summary key used for the default sort of summarized trace entries is shown in the first column. The summary key for the TSUML service is the first eight characters of location name.

Description:
Summarizes the traced transactions by location name. The display is a scrollable list of summary lines for each unique location name (eight characters only) found in the trace entries.

TSUMP—Trace Summary by Plan Name (PLAN)

The Trace Summary by Plan panel is a scrollable list of trace entries summarized and sorted by plan name (PLAN column). Both totals and averages of key workload indicators are shown (see "Display Fields" on page 165).

```
BMC Software --------------- TRACE SUMMARY BY PLAN --------------  RX AVAILABLE
SERV ==> TSUMP           INPUT    14:24:47  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SORT=PL,GRAPH=NO                      ROW 1 OF 5 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY                ENTRIES IN DATASET 1 - 132
 19MAR01                PBCRCP - SQL/SCAN/IO    19MAR01 - 17MAR01
           ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
 PLAN      COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
 -------- ------ -------- -------- ------ ------ -------- -------- ------ ------
 DSNESPCS      2 5,516 ms   307 ms    3.0   70.0    11 s    615 ms      6    140
 DSNESPRR      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     22 6,380 ms    18 ms    7.5   10.0 00:02:20   398 ms    165    220
                         *** END OF SUMMARY ENTRIES ***
```

Figure 56. Trace Summary by Plan Name

Select Code:

TSUMP

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace. A blank can be an identifier. This parameter is the first positional parameter; a comma delimits the next parameter. If there is no identifier, a comma must precede any other parameter.

,GRAPH=NO|AVG|TOT
: Displays a graphic presentation of the Entry Count (number of threads) followed by either the averages (AVG) or totals (TOT). NO requests a numerical display of both average and total values. Default is NO.

,SORT=PL|EC|AE|AC|A#|AG|TE|TC|T#|TG
: When the list of summarized trace entries is displayed initially, the list is sorted alphabetically by plan (default) and ,SORT=PL is displayed in the PARM field (the comma is a required positional parameter for SORT). SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 169.

"Selecting a Subset of Trace Entries" on page 171 provides more information about the following parameters that applies to all the trace summary services:

,A|AUTHID=xxxxxxxx
: Selects the specified authorization ID only

,C|CONNECT=xxxxxxxx
: Selects the specified connection name only

,P|PLAN=xxxxxxxx
: Selects the specified plan name only

,L|LOC=xxxxxxxx
: Selects the specified location name only

,R|CORRID=xxxxxxxxxxxx
Selects the specified correlation ID only

,T|TIME=hhmm[-hhmm]
Selects the specified time only

,D|DAY=nn[-nn]
Selects the specified day only

Summary Key:
PLAN is the first column of this display. The summary key used for the default sort of summarized trace entries is shown in the first column. The summary key for the TSUMP service is plan name.

Description:
Summarizes the traced transactions by plan name. The display is a scrollable list of summary lines for each unique plan name found in the trace entries.

TSUMR—Trace Summary by Correlation ID (CORR-ID)

The Trace Summary by Correlation ID panel is a scrollable list of trace entries summarized and sorted by the DB2 correlation ID (CORR column). Both totals and averages of key workload indicators are shown (see "Display Fields" on page 165).

```
BMC Software --------------TRACE SUMMARY BY CORR-ID -------------  RX AVAILABLE
SERV ==> TSUMR           INPUT    15:15:22  INTVL=> 5  LOG=> N  TGT==> DB2HHH
PARM ==> WORKSTTN,SORT=CR,GRAPH=NO                    ROW 1 OF 3 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC)                         ENTRIES IN DATASET 1 - 74
                              DB2 APPLICATION TRACE
          ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
CORR      COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
-------- ------ -------- -------- ------ ------ -------- -------- ------ ------
db2bp.ex     72    11 s    742 ms    9.1   14.6 00:13:47    53 s     658  1,049
BOLJEH1       1 01:19:13 9,184 ms    0.0    0.0 01:19:13 9,184 ms      0      0
BOLJEH2       1 00:42:42 00:04:15   29.0   295K 00:42:42 00:04:15     29   295K
************************** END OF SUMMARY ENTRIES **************************
```

Figure 57. Trace Summary by Correlation ID

Select Code:

TSUMR

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace. A blank can be an identifier. This parameter is the first positional parameter; a comma delimits the next parameter. If there is no identifier, a comma must precede any other parameter.

,GRAPH=NO|AVG|TOT
: Displays a graphic presentation of the ENTRY COUNT (number of threads) followed by either the averages (AVG) or totals (TOT). NO requests a numerical display of both average and total values. The default is NO.

,SORT=CR|EC|AE|AC|A#|AG|TE|TC|T#|TG
: When the list of summarized trace entries is displayed initially, the list is sorted alphabetically by DB2 correlation ID (default) and ,SORT=CR is displayed in the PARM field (the comma is a required positional parameter for SORT). SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 169.

"Selecting a Subset of Trace Entries" on page 171 provides information about the following parameters that applies to all the trace summary services:

,A|AUTHID=xxxxxxxx
: Selects the specified authorization ID only

,C|CONNECT=xxxxxxxx
: Selects the specified connection name only

,P|PLAN=xxxxxxxx
: Selects the specified plan name only

,L|LOC=xxxxxxxx
: Selects the specified location name only

,R|CORRID=xxxxxxxxxxxx
Selects the specified correlation ID only

,T|TIME=hhmm[-hhmm]
Selects the specified time only

,D|DAY=nn[-nn]
Selects the specified day only

Summary Key:
CORR is the first column of this display. The summary key used for the default sort of summarized trace entries is shown in the first column. The summary key for the TSUMR service is the DB2 correlation ID.

Note: This column displays only the first eight characters of the correlation ID; however, all twelve characters of the correlation ID are used and sorted on.

Description:
Summarizes the traced transactions by DB2 correlation ID. The display is a scrollable list of summary lines for each unique correlation ID found in the trace entries.

TSUMT—Trace Summary by Time (INTERVAL START)

The Trace Summary by Time panel is a scrollable list of trace entries summarized and sorted by time (INTERVAL START column). Both totals and averages of key workload indicators are shown (see "Display Fields" on page 165).

```
BMC Software -------------- TRACE SUMMARY BY TIME   -------------  RX AVAILABLE
SERV ==> TSUMT           INPUT    14:26:50  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,INTVL=1M,SORT=IS,GRAPH=NO             ROW 1 OF 3 SCROLL=> CSR
EXPAND:  LINESEL(LTRAC), HISTORY               ENTRIES IN DATASET 1 - 132
 19MAR01               PBCRCP - SQL/SCAN/IO    19MAR01 - 17MAR01
 INTERVAL  ENTRY    AVG      AVG     AVG    AVG    TOTAL    TOTAL   TOTAL  TOTAL
 START     COUNT  ELAPSED    CPU   #STMTS GETPGS  ELAPSED    CPU   #STMTS GETPGS
 -------- ------ -------- -------- ------ ------ -------- -------- ------ ------
 14:00:00     38 00:05:12   464 ms   16.2   67.3 03:17:59    17 s     616  2,560
 15:00:00     15    19 s    103 ms   34.9   45.0 00:04:50 1,550 ms    524    675
                         *** END OF SUMMARY ENTRIES ***
```

Figure 58. Trace Summary by Time

Select Code:

TSUMT

Parameter:

The following parameters can be specified:

trace id

Unique ID identifying the trace. A blank can be an identifier. This parameter is the first positional parameter; a comma delimits the next parameter. If there is no identifier, a comma must precede any other parameter.

,INTVL=nnM|nnH

This parameter allows a different time interval to be selected. When the list of summarized trace entries is displayed initially, the list is summarized by hour (default) and ,INTVL=1H is displayed in the PARM field (the comma is a required positional parameter for INTVL).

The summary time interval can be changed by typing nnM or nnH where

nnM

Specifies minutes

nnH

Specifies hours

The first INTERVAL START displayed is determined by rounding the earliest end time of a trace entry back to an even interval start time.

- Any minute interval that divides evenly into 60 starts at the latest prior period from an even hour. Those not evenly divisible start at the current minute.
- Any hour interval that divides evenly into 24 starts at the latest prior period from midnight. Those not evenly divisible start at the current hour.

  For example:

  – If an interval is 15 minutes and the earliest end time is 12:41:30, the first interval start time would be 12:30:00.

  – If an interval is 25 minutes and the earliest end time is 12:41:30, the first interval start time would be 12:41:00.

  – If an interval is 4 hours and the earliest end time is 9:21:35, the first interval start time would be 08:00:00.

,GRAPH=NO|AVG|TOT
Displays a graphic presentation of the Entry Count (number of threads) followed by either the averages (AVG) or totals (TOT). NO requests a numerical display of both average and total values. Default is NO.

,SORT=IS|EC|AE|AC|A#|AG|TE|TC| T#|TG|ISD
When the list of summarized trace entries is displayed initially, the list is sorted in ascending sequence by interval start (default) and ,SORT=IS is displayed in the PARM field (the comma is a required positional parameter for SORT). ISD (Interval Start Descending) displays the most current measurements at the top.

SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 169.

Note: You also can sort on each column by tabbing to the column header and pressing ENTER.

"Selecting a Subset of Trace Entries" on page 171 provides more information about the following parameters that applies to all the trace summary services:

,A|AUTHID=xxxxxxxx
Selects the specified authorization ID only

,C|CONNECT=xxxxxxxx
Selects the specified connection name only

,P|PLAN=xxxxxxxx
Selects the specified plan name only

,L|LOC=xxxxxxxx
Selects the specified location name only

,R|CORRID=xxxxxxxxxxxx
Selects the specified correlation ID only

,T|TIME=hhmm[-hhmm]
Selects the specified time only

,D|DAY=nn[-nn]
Selects the specified day only

Summary Key:

INTERVAL START is the first column of this display. The summary key used for the default sort of summarized trace entries is shown in the first column. The default summary key for the TSUMT service is the start timestamp of the summarized time interval.

If there is data for only one day in the trace buffer, the format of the timestamp is hh:mm:ss. If there is data for more than one day, the format is dd-hh:mm, where dd is the relative day number from the date of the first accounting record in the trace buffer.

Each accounting record traced is assigned to a time period based on its end timestamp. Therefore, the processing done for that unit of work is assigned to one interval, but work may have partially occurred during one or more previous time intervals depending on its elapsed time.

Description:

Summarizes the traced transactions by time interval start. The display is a scrollable list of summary lines by a start timestamp of each summarized time interval found in the trace entries.

Chapter 13. Viewing Summary Data for a Specific Thread

To view summary data for a specific thread, select an entry in the LTRAC display and press ENTER to access the STRAC display for that thread.

STRAC—Summary Trace Entry

The Summary Trace Entry panel displays summary data for a completed transaction or thread. There is one per DB2 accounting record. This panel is scrollable and displays a base section with the most critical information, followed by sections showing all accounting data. For a detail trace, additional summary sections may be available for SQL activity, database scans, database lock and I/O activity, and sort activity.

Note: If the target for a historical trace is not equal to the current TGT field, it is displayed in a HIST TGT field.

Figure 59–Figure 74 show representative displays of each STRAC section.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC           INPUT    15:14:16  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SEQ=2                                     ROW 1 OF 129 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, CURRENT                          HIST TGT--- DSN2
        ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, PKG, SPAS, DDF
        SUMMARIES:  SQL, SCANS, IO/LOCK, SORTS
STOP.....07MAY 13:35:20.20 PLAN...............DSN8CC0 TYPE..............ALLIED
START....07MAY 13:35:20.17 AUTHID.............T127 CONNECT.....LRTEST1/CICS
ELAPSED.............33 ms ORIG PRIM AUTH........T127 CORR ID.....GT00PROB
TERM......NORMAL/APPL END  COMMITS.................1 ROLLBACKS.............0
-------------------------------------------------------------------------------
RUNTIME ANALYSIS   IN DB2     IN APPL.      TOTAL    %IN DB2(=)     TOTAL(*)
----------------   --------   --------   --------   0 ...25...50...75..100%
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 (PRLL=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 59. Summary Trace Entry Panel (Base Section)

See “Thread Identifiers and Status Summary (Base Section)” on page 198.

```
- - - - - - - - - - - ENVIRONMENTAL INDICATORS - - - - - - - - - - - - - - -
LUWID...........USBOOL1.DSN2.A8E1BAF1B5DF0001
RLF TABLE ID....NOT ACTIVE
 - - - - - - - - - - - - - - - CICS CONNECTION - - - - - - - - - - - - - - -
ACCOUNTING TOKEN......(TOKENE NOT SPECIFIED)
```

Figure 60. STRAC Environmental (ENV) Section

See “Environmental Indicators” on page 210.

```
- - - - - - - - ELAPSED TIME ANALYSIS (ACCTG CLASSES 2,3 ONLY) - - - - - - - -

CATEGORY           #EVENTS AVG/EVENT  ELAPSED  %TOTAL
----------------   ------- --------- -------- ------- 0 ...25...50...75..100%
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 61. STRAC Elapsed Time Analysis (ELAPSED) Section

See “Elapsed Time Analysis” on page 211.

```
- - - - - - - - - - - - SQL STATEMENT EXECUTION COUNTS - - - - - - - - - - - -
SELECT...............0    LOCK TABLE..............0    CREATE............0
INSERT...............0    GRANT/REVOKE............0    DROP..............0
UPDATE...............0    SET CURR.SQLID..........0    ALTER.............0
DELETE...............0    SET HOST VAR. ..........0
                           SET CURR.DEGREE.........0    COMMENT ON........0
PREPARE..............1    SET CURR. RULES.........0    LABEL ON..........0
DESCRIBE.............0    CONNECT TYPE 1..........0
                           CONNECT TYPE 2..........0    CREATE TEMP.......0
OPEN.................1    SET CONNECTION..........0    RENAME TBL........0
FETCH................5    RELEASE.................0
CLOSE................1    CALL....................0
                           ASSOCIATE LOCATOR.......0
                           ALLOCATE CURSOR.........0

** DML................8    ** DCL..................0    ** DDL............0
** REOPTIMIZED........0
                   - - - - DYNAMIC SQL CACHE COUNTS - - - -
STMT FOUND...........0    KPDYN-PREP.AVOID........0    STMT DISCRD.......0
STMT NOT FOUND.......0    KEEPDYN_IMPL.PREP.......0    STMT PURGED.......0
```

Figure 62. STRAC SQL Statement Execution Counts (SQLCOUNTS) Section

See “SQL Statement Execution Counts” on page 216.

```
- - - - - - - - - - - - - - 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
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
NBR UNREG PAGES....         1         -         1
NBR EXPLICIT XI....         0         -         0
NBR WRITES 2ND BP..         0         -         0
```

Figure 63. STRAC Buffer Pool Activity (BPOOL) Section

See “Buffer Pool Activity” on page 220.

```
- - - - - - - - - - - - - - - LOCK ACTIVITY  - - - - - - - - - - - - - -  - -
TIMEOUT.........................0     MAX PAGE/ROW LOCKS...............1
DEADLOCK........................0
                                      LOCK REQUEST....................6
SUSPEND-LOCK....................0     UNLOCK REQUEST..................9
SUSPEND-LATCH...................0     QUERY REQUEST...................0
SUSPEND-OTHER...................0     CHANGE REQUEST..................8
                                      OTHER REQUEST...................0
ESCALATION(SHR).................0
ESCALATION(EXCL)................0

DRAIN REQUEST...................0     DRAIN FAILURE...................0
CLAIM REQUEST...................2     CLAIM FAILURE...................0
                    - - - - GLOBAL LOCK ACTIVITY - - - -
LOCK REQUEST....................1     XES LOCK REQUEST................3
LOCK CHANGE REQUEST.............0     XES CHANGE REQUEST..............0
UNLOCK REQUEST..................0     XES UNLOCK REQUEST............. 1

IRLM GLOBAL SUSPEND.............0     XES GLOBAL SUSPEND..............0
FALSE CONTENTION................0

INCOMPATIBLE RETAINED...........0     NOTIFY MESSAGES SENT............0
```

Figure 64. STRAC Lock Activity (LOCKS) Section

See “Lock Activity” on page 223.

```
- - - - - - - - - - - - - - - - PARALLELISM  - - - - - - - - - - - - - - - - - -
MAXIMUM DEGREE................2        FALLBACK - NO BUFFER............1
GROUPS EXECUTED...............3        FALLBACK - NO ESA SORT..........0
-- PLANNED DEGREE.............1        FALLBACK - AMBIG. CURSOR........1
-- REDUCED DEGREE.............2

PARALLEL TASKS................3        FALLBACK - NO ENCLAVE...........0
CORRELATION TOKEN.....X'00000000'       FALLBACK - RLF LIMITED.........NO
                          - - - - SYSPLEX - - - -
COORDN/ASSISTANT...........COORDN       FALLBACK - NO COORDN............1
PRLL COORDN NAME...............         FALLBACK - ISOL RR/RS...........2
MAXIMUM MEMBERS...............2        REDUCED  - VP SHORTAGE..........0
INTENDED PRLL GROUPS..........0
QUERY REFORMULATED (ENV)......0        QUERY REFORMULATED (BP).........0
```

Figure 65. STRAC Parallelism (PRLL) Section

See “Parallelism” on page 226.

```
- - - - - - - - - - - - - - -STORED PROCEDURES- - - - - - - - - - - - - - - -
SPAS CPU TIME..............200 us        SQL CALLS.......................9
SPAS IN-DB2 TIME...........200 us        CALLS ABENDED...................0
SPAS WAITING ON TCB........600 ms        CALLS REJECTED..................0
                                         CALLS TIMED OUT.................0
Total elapsed time.........200 us        WAITS ON TCB....................6
Elapsed time in SQL........200 us
                                         SQL nest level..................5

---and/or--

- - - - - - - - - - - - USER-DEFINED FUNCTIONS- - - - - - - - - - - - - - - -
UDF CPU TIME...............200 us        UDFs executed...................9
UDF IN-DB2 TIME............200 us        UDFs ABENDED....................0
UDF WAITING ON TCB.........600 ms        UDFs REJECTED...................0
                                         UDFs TIMED OUT..................0
Total elapsed time.........200 us
Elapsed time (SQL).........200 us        SQL STMTS executed.............15
                                         SQL nest level..................5

---and/or--

- - - - - - - - - - - - - - - - TRIGGERS  - - - - - - - - - - - - - - - - - -
-- non-enclave times --                  Stmt TRG activated..............9
CPU Time...................200 us        Row TRG activated...............0
Elapsed time...............600 ms        Trigger SQL errors..............0
-- enclave times --
CPU time...................200 us        CPU time (before enclave)...200 us
Elapsed time...............200 us
                                         SQL nest level..................5
```

Figure 66. STRAC Routines (RTN) Section

See “Routines” on page 229.

```
- - - - - - - - - - - - - - - - - - DDF SUMMARY - - - - - - - - - - - - - - - - -
CURRENT LOCATION: DB2D (DBAT SERVER/ALLIED DIST REQUESTOR)
   WORKSTATION USER ID...bolbpl2          WORKSTATION NAME..
   WORKSTATION TRANSACTION ID...db2bp.exe

   MVS WLM SERVICE CLASS NAME......DB2JWLM

REMOTE            FROM         TO
  LOCATION:  DB1D                        SQL STATEMENTS           0          1
  REMOTE PROD-ID:    DB2PROD             ROWS                   250          0
  DB AGENT CPU............505 ms        BYTES                63,237      1,237
  ELAPSED LOCAL.........00:02:19        TRANSACTIONS             0          1
  ELAPSED REMOTE........00:02:18        MESSAGES                 5          6
  CONVERSATIONS QUEUED......0           CONVERSATIONS            0          1
  BLOCK FETCH %............21%         COMMITS                  0          1
  MAX CONVERSATIONS.........1           ABORTS                   0          0
  UNSUCCESSFUL CONV'S.......0           BLOCK FETCH             20          0

TWO-PHASE COMMIT (2PH):                 2PH PREPARE              0          0
                                        2PH LAST AGENT           0          0
REMOTE COORDINATOR                      2PH COMMITS              0          0
------------------                      2PH BACKOUTS             0          0
-- INDOUBT................1            2PH FORGET RSP           0          0
-- COMMIT.................3            2PH REQ COMMIT           0          0
-- ROLLBACK...............0            2PH BACKOUT RSP          0          0
```

Figure 67. STRAC DDF Summary (DDF) Section

See “DDF Summary” on page 233.

```
- - - - - - - - PACKAGE / DBRM OVERVIEW (ACCTG CLASSES 7,8 ONLY) - - - - - - -

PACKAGE/                                          %TOTAL
DBRM      NO. SQL  CPU TIME  WAIT TIM  ELAPSED   ELAP.
-------- --------  --------  --------  -------- ------ 0 ...25...50...75..100
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 68. STRAC Package/DBRM Overview (PKG) Section

See “Package/DBRM Overview” on page 237.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> STRAC           INPUT    11:12:21  INTVL=> 3  LOG=> N  TGT==> DB2E
PARM ==> THRDHIST,SEQ=6                                          SCROLL=> CSR

- - - - - - - - PACKAGE / DBRM SECTION (ACCTG CLASSES 7,8 ONLY) - - - - - - -

NAME:            DB2E.BOLLA1.ACUT0300
TYPE:            PACKAGE                LOCATION:           DB2E
PROGRAM:         ACUT0300               COLLECTION ID:      BOLLA1
    LOADED BY A STORED PROCEDURE        CONSISTENCY TOKEN:  152B76830BA4B826

SQL STATEMENTS -  TOTAL:        8
DB2 ENTRY/EXIT -  TOTAL:       29

TIMINGS            #EVENTS AVG/EVENT   IN-DB2  %TOTAL
-----------------  ------- --------- -------- ------- 0 ...25...50...75..100
ELAPSED TIME                           139 ms    0.15 | <                    |
TCB CPU TIME                         7,831 us    0.00 |                     |
TOTAL WAITS:            14 8,250 us    116 ms    0.13 | <                    |
  LOCK/LATCH             0     0 us      0 us    0.00 |                     |
  I/O WAIT               8 4,451 us     36 ms    0.04 | <                    |
  OTHER READ             0     0 us      0 us    0.00 |                     |
  OTHER WRITE            0     0 us      0 us    0.00 |                     |
  UNIT SWITCH            6    13 ms     80 ms    0.09 | <                    |
  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 |                     |
  GLOBAL LOCK            0     0 us      0 us    0.00 |                     |
  DATA SHR MSGS.         0     0 us      0 us    0.00 |                     |
  UDF Schedule           0     0 us      0 us    0.00 |                     |
*NOT ACCOUNTED                         307 us    0.00 |                     |
```

Figure 69. STRAC Package/DBRM Pop-up Display

See “Package/DBRM Pop-up” on page 239.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC  INPUT    14:27:16  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,SQL,SORT=PGM    ROW 1 OF 18 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, 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+
------- -----  -----  -------- -----  -------- -----  ----  ---- ---- ----
SELECT   3228      1     25 ms   2.7  1,930 us   0.3     0     2    1    0
SELECT   3347      2     11 ms   2.4  4,047 us   1.5     0    11    4    0
OPEN     3565      6    172 us   0.1    170 us   0.2     0     0    0    0
FETCH    3578     11  1,986 us   2.3    782 us   1.6     0    19    3    0
CLOSE    3664      6    192 us   0.1    149 us   0.2     0     0    0    0
SELECT   3671      6  1,776 us   1.1  1,091 us   1.2     0    12    0    0
SELECT   3283      1  2,643 us   0.3  1,888 us   0.3     0     1    0    0
OPEN     3299      1    109 us   0.0    107 us   0.0     0     0    0    0
FETCH    3313      2    907 us   0.2    855 us   0.3     0     1    0    0
CLOSE    3334      1    142 us   0.0    141 us   0.0     0     0    0    0
OPEN     3456      1    693 ms  74.1    414 ms  74.9     8    23 1187    2
FETCH    3468      5    440 us   0.2    322 us   0.3     0     0    0    6
SELECT   4803      1  1,269 us   0.1  1,121 us   0.2     0     2    0    0
PGM: P025D100     44            83.9            81.0     8    71 1195    8

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

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

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

Figure 70. STRAC SQL Summary Section

See “SQL Summary” on page 242.

```
BMC SOFTWARE---------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> STRAC           INPUT    10:22:51  INTVL=> 3  LOG=> N  TGT==> DB2C
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                              ----------------------------
                                                   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
LOB- PAGES SCANNED                                     0         0         0
   - PAGES UPDATED                                     0         0         0
```

Figure 71. STRAC SQL Statement Pop-up Display

See “SQL Statement Pop-up” on page 246.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC  INPUT    14:27:46  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,SCAN,SORT=DA    ROW 1 OF 11 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, PKG, SPAS, DDF
         SUMMARIES: SQL, SCANS, IO/LOCK, SORTS
- - - - - - - - - - - DATABASE SUMMARY (DETAIL TRACE ONLY) - - - - - - - - - -

                        ------ INDX -----  ------ DATA -----  ------ WORK+ ----
DATABASE  PAGESET OBID  #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 72. STRAC Database Summary Section

See “Database Summary” on page 249.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC  INPUT    14:27:53  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,IOLOCK,SORT=DP  ROW 1 OF 11 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, 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
-------- -------- ----- ----- ------  ----- ----- -------- -------- ----- -----
B1C0DB   BB040A12     0     0      0      1     0    21 ms    21 ms     0     0
B2C0DB   BB051012     2     0      0      1     0    18 ms    18 ms     0     0
B3C0DB   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 73. STRAC Database Lock and I/O Summary Section

See “Database Lock and I/O Summary” on page 251.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC           INPUT    14:27:16  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,SORTS                          ROW 1 OF 18 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, 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 74. STRAC Sort Summary Section

See “Sort Summary (Detail Trace Only)” on page 253.

Select Code:

STRAC

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace. A blank can be an identifier. Therefore, this parameter is positional and a comma delimits the next parameter. For example, if there is no identifier, a comma must precede any other parameter. However, if there is an ID, it must be in the first position.

,SEQ=
: A unique internal sequence number assigned to each trace entry. The number is assigned sequentially throughout the trace, across multiple log data sets. If this keyword is not entered, the first entry in the trace buffer is displayed. The value after the equals sign can be overtyped to jump directly to any valid trace entry.

,ENV|ELAPSED|SQLCOUNTS|BPOOL|LOCKS| PRLL|DDF|PKG|SPAS
: A request to display a specific STRAC accounting section as described on page 196.

,SQL|SCAN|IOLOCK|SORTS
: A request to display a specific STRAC summary section as described on page 197.

,SORT=
: A request to sort a qualified SQL or Database Summary display as described in “SQL Summary” and “Database Summary” in this section.

Description:

Displays accounting data on the activity generated by one transaction or thread. It also may include summaries of detail trace events. It is invoked by cursor selection from the LTRAC scrollable list display.

Note: All the fields are formatted from the DB2 accounting record that is written by DB2 to the specified destination. A record is written for each DB2 transaction and is available to the trace facility when the transaction terminates. For more information, refer to the IBM publication, *DB2 Administration Guide*.

Scrolling:

By using the PF10/22 and PF11/23 keys, you can scroll backwards or forwards one trace buffer entry (transaction) at a time. The current entry number is displayed in the SEQ= field. PF7/19 scrolls the display up and PF8/20 scrolls the display down.

Expand:

The STRAC display can be EXPANDed to the following displays:

MON(WKLD)
: Active Timer Requests display of all active monitors in the DB2 workload (WKLD) area

DETAIL
: Detail Trace Entry display

HISTORY
: Historical trace data set for this trace (if you are viewing a current trace with logging, this provides access to data no longer in the online buffer).

CURRENT
: Active trace data still in the buffers (if you are viewing a trace log data set for a still-active trace or a complete trace that has not yet been purged).

ACCOUNTING:
: The following sections format data from the DB2 accounting record:

ENV
: STRAC Environmental Indicators display section

ELAPSED
: STRAC Elapsed Time Analysis display section (accounting classes 2 and 3 only)

SQLCOUNTS
: STRAC SQL Statement Execution Counts display section

BPOOL
: STRAC Buffer Pool Usage Analysis display section

LOCKS
: STRAC Lock Activity display section

PRLL
: STRAC Parallelism display section

RTN
: STRAC Routines display section (available only if the transaction has issued stored procedure calls or user-defined functions or has been invoked by a trigger)

DDF
: STRAC DDF Summary display section (available only if there is DDF activity for the displayed thread)

PKG
: STRAC Package/DBRM Overview display section (available only when accounting trace 7 is active)

From the Package/DBRM Overview section of STRAC, you can expand to:

Package/DBRM Pop-up
: Place the cursor on a line of data for a package or DBRM in the Package/DBRM Overview section of STRAC and press ENTER to view a pop-up display of detail statistics for that package or DBRM. See "Package/DBRM Pop-up" on page 239 for a description of this pop-up.

SUMMARIES:

The following sections format data summarized from detail trace events and are available only if these events are traced.

SQL

STRAC SQL Summary display section

From the SQL Summary section of STRAC you can expand to these two displays:

SQL Statement Pop-up

Place the cursor on a line of data for an SQL statement in the SQL Summary section of STRAC and press ENTER to view a pop-up display that summarizes all statistics from the individual occurrences of that statement. See "SQL Statement Pop-up" on page 246 for a description of this pop-up.

DTRAC PGM=

Place the cursor on the PGM line in the SQL Summary section of STRAC and press ENTER to request a DTRAC display of only those events from a given program.

The statements displayed by DTRAC are qualified by the program name specified by the PGM field. Only those SQL statements issued by that originating program and all events subordinate to the selected SQL statements are shown. For more information about the DTRAC display, see "DTRAC—Detail Trace Entry" on page 255.

SCANS

STRAC Database Summary display section

IO/LOCK

STRAC Lock and I/O Summary display section

SORTS

STRAC Sort Summary display section

You also can tab to any one of several defined lines and press ENTER to transfer to the following related displays:

Line Identifier	STRAC Section Transferred To
CURRENT	ENV
ELAPSED TIME	ELAPSED
TOTAL SQL	SQLCOUNTS
GETPAGES	BPOOL
SYNC I/O (PRLL=nn)	PRLL
UPDATES/COMMIT	LOCKS

Any other position transfers to DTRAC.

Thread Identifiers and Status Summary (Base Section)

The following areas are displayed on the initial screen where the most critical data is summarized.

Area 1 - Identifiers and Status

The initial screen summarizes the most critical data. This area shows timestamps, status, and identifiers.

```
STOP.....07MAY 13:35:20.20 PLAN.............TECPRPLN TYPE.............ALLIED
START....07MAY 13:35:20.17 AUTHID............FRIEDMK CONNECT.....LRTEST1/CICS
ELAPSED.............33 ms ORIG PRIM AUTH.....FRIEDMK CORR ID.....GT00PROB
TERM......NORMAL/APPL END  COMMITS.................1 ROLLBACKS.............0
```

where

STOP
: Stop date and time of transaction.

START
: Start date and time of transaction.

ELAPSED
: Elapsed time for this task. If accounting trace level one is not active, this is the total elapsed time to date for this task.

  For TSO, it is the elapsed transaction time (since the ENTER key was pressed). If accounting trace level one is active, this is the elapsed time since this thread was established at the plan level.

  For CICS and IMS, it is the time since the transaction thread issued its first SQL. If accounting trace level one is active, this is the elapsed time since this thread was established at the plan level.

TERM
: Indicates the termination reason for the transaction. This field consists of two parts. The first part shows the major status indicator, which can be one of the following:

NORMAL	Normal termination.
ABNORM	Abnormal termination.
IN DBT	Indoubt termination.
INCMP	The transaction has not yet terminated. Data is incomplete in the trace log data set being viewed. Only detail data is available; no DB2 accounting statistics exist.

The second part shows the detail status. Possible combinations are as follows:

NORMAL/IFI READ
NORMAL/RESIGNON
NORMAL/NEW USER
NORMAL/DBAT INACT
NORMAL/DEALLOC
NORMAL/APPL END

ABNORM/EOT-APPL ABN
ABNORM/EOM-END MEM
ABNORM/R IN DOUBT
ABNORM/STOP FORCE

IN DBT/APPL TERM
IN DBT/EOT-APPL ABN
IN DBT/EOM-END MEM
IN DBT/STOP FORCE

PLAN
Plan name

AUTHID
Authorization ID

ORIG PRIM AUTH
Original primary authorization ID

COMMITS
Number of commits

TYPE
The type of thread, which can be one of the following:

ALLIED	Standard thread
ALLIED DISTR	Thread requesting distributed data (requestor)
DBAT	Database access thread (distributed server)

CONNECT

The subsystem attach name from where this user is making DB2 requests. The connection name is followed by the connection type where available.

TSO/TSO	TSO Foreground and Background
BATCH	Batch, no type specified
UTILITY	Utility, no type specified
DB2 CALL/CAF	Call Attach Facility
cicsid/CICS	CICS Attach
imsid/MPP	IMS Attach—MPP
imsid/BMP	IMS Attach—BMP
imsid/TBMP	IMS Attach—Transaction BMP
imsid/CTL	IMS Attach—Control Region
SERVER/SYS	Distributed, System-Directed Access
SERVER/APL	Distributed, Application-Directed Access
db2id	DB2 Internal, no type specified
jobname/DLIB	DL/I Batch
xxxxx/SYS	Distributed, System-Directed Access, where xxxxx is the connection name of the thread at the requesting location
jobname/TRRS	RRSAF (Recoverable Resource Manager Services Attachment Facility) (DB2 version 5 and later)
jobname/UTIL	DB2 utilities subtask connection (DB2 version 5 and later)

The connection type determines the contents of the thread ID, authorization ID, and correlation ID fields.

CORRID

Correlation ID

ROLLBACKS

The number of implicit and explicit ROLLBACKs performed.

Area 2 - Runtime Analysis

This area shows runtime analysis of elapsed and CPU time with a graphic display.

```
-------------------------------------------------------------------------------
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                            | <                     |
```

where

ELAPSED TIME:

TOTAL
: Total elapsed time. The percent of total elapsed time is always 100 percent. This measure is included to establish the scale.

IN DB2
: Elapsed time spent in DB2 processing (accounting level 2).

IN APPL.
: Total elapsed time minus the time spent in DB2 (accounting level 2).

CPU TIME:

TOTAL
: TCB time minus the thread start time for the thread subtask processing the user's request.

  For CICS, this is a relative cost for the DB2 activity for this transaction. Any CICS CPU time for this transaction is not included in this value. In the event of thread reuse, this value can be distorted, as it is a total time for the thread subtask, not per CICS transaction.

IN DB2
: CPU time spent in DB2 processing (reported by DB2 only if accounting level 2 trace is active).

IN APPL.
: Total CPU time minus the time spent in DB2 (accounting level 2).

DB2 WAIT TIME
: Elapsed time spent in DB2 processing. (This field is zero unless accounting level 3 is active.)

Note: It is possible under some circumstances that DB2 wait time can exceed total elapsed time. This is possible because DB2 adds all wait times for all processes. If the waits for concurrent processes overlap, this time is counted multiple times.

Area 3 - Activity

This area shows the most important activity indicators.

```
- - - - - - ACTIVITY - - - - - -
TOTAL SQL.......................12
GETPAGES........................21
SYNC READS (PRLL=00)............0
PREFETCH PAGES READ.............1
UPDATES/COMMIT................0.0
BFR HIT RATIOS:...VP=100%,HP=  0%
```

where

TOTAL SQL
: Total number of SQL statements executed.

GETPAGES
: Number of GETPAGE requests. This field counts conditional and unconditional requests (both successful and unsuccessful.)

SYNC READS
: Number of synchronous read I/Os performed on behalf of this user.

(PRLL=nn)
: Number of parallel groups that were executed.

PREFETCH PAGES READ
: Total number of pages read by all prefetch performed for this request. This value includes all pages read by list, sequential, and dynamic prefetch.

UPDATES/COMMIT
: Number of SQL update type statements (INSERTs, UPDATEs, DELETEs) per commit statement.

BFR HIT RATIOS:
: Hit ratios for both the virtual pool and the corresponding hiperpool.

VIRTUAL POOL HIT RATIO
: Measures buffer pool performance. A high hit ratio means that a high percentage of reads are satisfied from the buffer pool without having to access external storage.

Note: Heavy prefetch activity can drive this value close to 0.

A pure random hit ratio cannot be calculated since the getpage count in the DB2 accounting record includes both random and sequential activity.

The hit ratio is computed as follows:

```
((A–B) / A) x 100
```

where

A Is the number of GETPAGE requests

B Is the number of synchronous reads and asynchronous pages read

HIPERPOOL HIT RATIO

Measures performance of the hiperpool. A high hit ratio means that a high percentage of reads are satisfied from the hiperpool. A hit ratio of 100% indicates that MVS has not stolen any of the expanded storage allocated to the hiperpool.

The hit ratio is computed as follows:

```
((A–B) / A) x 100
```

where

A Is the total number of successful reads from the hiperpool

B Is the number of hiperpool read failures

Area 4 - Key Indicators

This area shows the six most important key indicators of failures or degradation.

```
- - - - - KEY INDICATORS  - - - - - - -
SQL: DYNAMIC(PREPARE)=     1
I/O RSP: SYNC=    45 ms, ASYNC=   107 ms
```

A maximum of six key indicators are displayed in this area. They are selected in the following order of importance:

- These indicators show the cause of degradation and are highlighted:
  - **TIMEOUT (or DEADLOCK)**

    Indicates that this unit of work failed because either a timeout condition or deadlock condition was detected.

    TIMEOUT means that the unit of work was suspended for a length of time longer than permitted by installation options.

    DEADLOCK means that two units of work attempted to access the same resource and prevented each other from completing normally.

    Action: N/A

    Field: QTXATIM, QTXADEA

  - **STORED PROC. FAILED (reason) = nnnnnn**

    Indicates the number of times a stored procedure call failed for one of the following reasons:

ABEND	The stored procedure call terminated abnormally.
REJECT	The stored procedure call was rejected.
TIMEOUT	The stored procedure call was timed out while waiting to be scheduled.

    Action: N/A

    Field: QXCALLAB, QXCALLRJ, QXCALLTO

  - **FILTER: fieldname=xxxxx (>n)**

    Indicates that this trace was selected because one of the trace exception filter conditions was met. These filter conditions are specified when the trace is started.

    where

fieldname	Name of the trace exception filter condition
xxxxxx	Target value
nnnnnn	Actual value measured

    See “Selecting Trace Data for Retention” on page 18.

    Action: N/A

    Field: N/A

– **RID FAILURE - STORAGE**

Indicates that RID list processing for this unit of work failed because not enough storage was available.

Action: Review the storage allocations specified when DB2 was installed.

Field: QXNSMIAP

– **RID FAILURE - NUMBER OF RIDS**

Indicates that RID list processing for this unit of work failed because one of the internal limits was exceeded.

The internal limits include the physical limit of the number of RIDs a RID list can contain and internal thresholds for the retrieval and manipulation of RIDs.

Action: N/A

Field: QXMRMIAP

– **CLAIM FAILURE**

Indicates that CLAIM processing failed for this unit of work.

Action: N/A

Field: QTXACLUN

– **DRAIN FAILURE**

Indicates that DRAIN processing failed for this unit of work.

Action: N/A

Field: QTXADRUN

– **PARALLEL CPU FALLBACK - reason**

Indicates that parallel CPU processing was turned off and sequential mode was used.

The possible causes indicated by *reason* are as follows:

NO ENCLAVE	The MVS ENCLAVE feature is not supported on this machine (MVS 5.2 or higher).
RLF	The Resource Limit Facility has limited CPU parallelism for some SELECT statement.

Action: N/A

Field: QXDEGENC, QXRLFDPA

– **PARALLEL I/O FALLBACK - reason**

Indicates that parallel I/O processing was turned off and sequential mode was used.

The possible causes indicated by *reason* are as follows:

STORAGE	Storage shortage or insufficient buffers in the buffer pool.
ESA SORT	The ESA sort feature is not installed on this machine.
AMBIG. CURSOR	The cursor in use is capable of UPDATE or DELETE.

Action: N/A

Field: QXDEGBUF, QXDEGESA, QXDEGCUR

– **SYSPLEX PRLL FALLBACK - reason**

Indicates that sysplex parallel processing was disabled and the parallel group was executed on a single DB2 (DB2 version 5 and later only).

Possible causes indicated by *reason* are as follows:

COORDINATR=NO	Bound COORDINATOR=YES, but run on COORDINATOR=NO system.
ISOLAT..RR/RS	PLAN/PACKAGE was bound with isolation level of repeatable read.
BYPASS-BUFFER	Bypassed due to insufficient buffer pool storage.

Action: N/A

Field: QXCOORNO, QXISOOR, QXXCSKIP

– **PARALLEL I/O REDUCED DEGREE - nnnnnn**

Indicates that parallel I/O processing was reduced from the planned degree to a lesser degree because of insufficient storage or insufficient buffers in the buffer pool.

where

nnnnnn Number of times this has occurred

Action: If this number is not zero, consider increasing the size of the buffer pool or specifying a different buffer pool for the table space.

Field: QXREDGRP

– **INCREMENTAL BINDS = nnnnnn**

Indicates the number of incremental binds this unit of work performed.

This indicates that the plan/package had become invalid for some reason, such as an ALTER on an index or table.

Action: Nothing needs to be done for this plan, but you may want to explicitly bind any plans whose tables/indexes are being altered. Incremental binds can have a significant impact in an online transaction system.

Field: QXINCRB

– **TOTAL DDL = nnnnnn**

Indicates the number of Data Definition Language (DDL) statements performed by this unit of work.

DDL statements, such as CREATE TABLE, can have a significant impact on system performance. To perform DDL operations, exclusive locks need to be obtained against the DB2 catalog.

Action: In an online transaction system, avoid designing transactions that use DDL operations.

Field: QXCRTAB, QXCRINX, and so on

– **BUFFER INCOMPLETE**

Indicates the transaction spans multiple trace log data sets. No accounting statistics data is available except in the last log, and detail trace data is incomplete in each log.

However, accounting data in the final log is complete and represents the entire transaction. The detail trace data in each log can be added together for total results.

Action: Allocate a larger trace log data set.

Field: N/A

- The following indicators are informational in nature and are not highlighted:

– **GRANTS/REVOKES = nnnnnn**

Indicates the number of GRANT and/or REVOKE SQL statements issued by this unit of work.

Action: N/A

Field: QXGRANT, QXREVOK

– **LOCK TABLE = nnnnnn**

Indicates the number of LOCK TABLE SQL statements issued by this unit of work.

Action: Use LOCK TABLE carefully. It means that this unit of work owns exclusive control of the table, allowing no concurrent access.

Field: QXLOCK

– **LOCK ESCALATIONS = nnnnnn**

Indicates the number of times that the total locks per table have exceeded the installation limit for this unit of work. When this limit is exceeded, DB2 promotes the current page locks to a single table space lock of the same type.

Action: Lock escalation usually indicates that the application has encountered an exceptional condition. If this occurs frequently, the design of the application should be reviewed.

Field: QTXALES, QTXALEX

– **PARALLEL I/O CONDITIONAL GETPAGE FAILURE**

Indicates that a conditional GETPAGE request was not satisfied for this buffer pool. This applies only to parallel I/O that is conditional in nature.

When this condition occurs, it indicates that a prefetch operation failed to retrieve a page before the application required it.

Action: N/A

Field: QBACNGT

– **SQL: SELECT=nnnnnn, FETCH=nnnnnn**

Summary of the number of SQL SELECT and FETCH statements issued by this unit of work.

In a DDF environment, the number of FETCH statements may not truly reflect the number of FETCHes issued by the application due to internal DDF processing.

Action: N/A

Field: QXSELECT, QXFETCH

– **SQL: INS=nnnnnn, UPD=nnnnnn, DEL=nnnnnn**

Summary of the number of SQL INSERT, UPDATE, and/or DELETE SQL statements issued by this unit of work.

Action: N/A

Field: QXINSRT, QXUPDTE, QXDELET

– **SQL: DYNAMIC(PREPARE)=nnnnnn**

Summary of the number of PREPARE SQL statements issued by this unit of work. Each PREPARE statement is equivalent to one dynamic SQL statement.

In a DDF environment, the number of PREPARE statements may not truly reflect the number of dynamic SQL statements issued by the application due to internal DDF processing.

Action: N/A

Field: QXPREP

– **I/O RSP: SYNC= time, ASYNC= time**

Provides an indication of the average I/O response times this unit of work experienced for synchronous and asynchronous requests.

Action: N/A

Field: For synchronous requests:
QWACARNE, QWACAWTI

For asynchronous requests:
QWACARNR, QWACARNW, QWACAWTR, QWACAWTW

– **LOCK SUSPENSIONS = nnnnnn**

Number of times this unit of work was suspended due to a lock or latch conflict with another unit of work.

This number should be small, ideally zero.

Action: N/A

Field: QTXASLOC, QTXASLAT, QTXASOTH

– **PARALLEL MAINTASK, SUBTASKS = nnnnnn**

Indicates that this unit of work is the primary, or originating unit of work in a group supporting a query or utility using parallel tasks. SUBTASKS shows the number of subtask units of work that were created to support this query or utility.

Action: N/A

Field: QWACPACE, QWACPCNT

– **PARALLEL - SUBTASK**

Indicates that this unit of work was created in support of a query or utility using parallel tasks.

Action: N/A

Field: QWACPACE

– **SYSPLEX PARALLELISM - COORDINATOR/ASSISTANT**

Indicates that a query was processed across more than one DB2, where this DB2 was either the coordinator or an assistant (DB2 version 5 and later only).

Action: N/A

Field: QWDA

– **NUMBER OF DISTRIBUTED LOCATIONS = nnnnnn**

Number of separate DDF locations accessed by this unit of work.

Action: N/A

Field: N/A

– **RID LIST PROCESSING USED = nnnnnn**

Number of times RID list processing was used for this unit of work.

During RID list processing, DB2 produces a list of candidate record IDs from an index. The resulting RID list can be used to efficiently retrieve the qualifying rows.

Action: N/A

Field: QXMAIP

– **RLF(nn), TYPE=xxxxxxxxxxxxxxx**

Indicates that the Resource Limit Facility (RLF) is active and displays the RLF limit rule that applies to this unit of work.

where

nn	Suffix of the Resource Limit Facility (RLF) table in effect.
xx..xx	RLF rule that applies to this transaction. Refer to the *DB2 Administration Guide* for the definition of the rule.

Action: N/A

Field: QTXARLID, QTXAPREC

Accounting Sections

The following sections format data from the DB2 accounting record.

Environmental Indicators

The Environmental Indicators section shown in Figure 75 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the ENV expand button
- Tabbing to the STOP field in the base section and pressing ENTER

```
- - - - - - - - - - - ENVIRONMENTAL INDICATORS - - - - - - - - - - - - - - -
LUWID...........USBOOL1.DSN2.A8E1BAF1B5DF0001
RLF TABLE ID....NOT ACTIVE
 - - - - - - - - - - - - - - - CICS CONNECTION - - - - - - - - - - - - - - -
ACCOUNTING TOKEN......(TOKENE NOT SPECIFIED)
```

Figure 75. STRAC Environmental (ENV) Section

where

LUWID
: Logical unit of work ID.

RLF TABLE ID
: Resource Limit Facility (RLF) table ID. If the Resource Limit Facility is inactive, INACT is shown instead of a table ID.

CICS CONNECTION
: Displayed only if the thread represents a CICS user. It is blank if this thread represents an IMS user, a batch job, Call Attach Facility (CAF), or TSO user.

ACCOUNTING TOKEN
: CICS correlation token, made up of network name, LU name, and uniqueness value.

Elapsed Time Analysis

The Elapsed Time Analysis section shown in Figure 76 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the ELAPSED expand button
- Tabbing to the ELAPSED TIME field in the base section and pressing ENTER

The data is displayed only if accounting class 2 or 3 is active. If accounting class 2 or 3 is not active, only the header is displayed.

```
- - - - - - - - ELAPSED TIME ANALYSIS (ACCTG CLASSES 2,3 ONLY) - - - - - - - -

CATEGORY           #EVENTS AVG/EVENT  ELAPSED  %TOTAL
----------------   ------- --------- -------- ------- 0 ...25...50...75..100%
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 76. STRAC Elapsed Time Analysis (ELAPSED) Section

This area displays the elapsed time, number of events, average elapsed time per event, and percentage of total elapsed time for each of the following categories. The percentage of total elapsed time is also displayed graphically for each of these categories.

ELAPSED TIME

This information is available if accounting class 2 is active:

IN DB2

Elapsed time spent in DB2 processing (QWACASC).

In-DB2 time is elapsed time while performing DB2 work. The DB2 time is a percentage of the total elapsed time.

Note: DB2 combines elapsed times of related processes into a single in-DB2 elapsed time value; thus the in-DB2 elapsed time may exceed the total elapsed time. When this situation occurs, the percent of total elapsed time is set to 101 percent.

IN APPLICATION

Total elapsed time minus the time spent in DB2 (QWACESC - QWACBSC - QWACASC).

Application time is elapsed time spent in the application while NOT performing DB2 work.

Note: DB2 combines elapsed times of related processes into a single in-DB2 elapsed time value; thus the in-DB2 elapsed time may exceed the total elapsed time. When this situation occurs, the in-application time is not calculated and is displayed as asterisks.

TOTALS

Total elapsed time from the first connect to DB2 to thread termination (QWACESC - QWACBSC).

The time a thread was inactive waiting for work is included in this value.

This value is not accurate for threads that do not terminate, such as a CICS conversational transaction or an IMS WFI BMP.

Note: The percent of total elapsed time is always 100 percent. This measure is included to establish the scale.

WAITS IN DB2 (LOCAL)

The elapsed time for the following DB2 wait events is available only if accounting class 3 is active, as follows:

LOCK/LATCH

Elapsed time spent in DB2 waiting for a lock or latch (QWACAWTL).

This value indicates the elapsed time the thread waited for transaction locks and latches.

This time reflects the amount of time the thread was suspended because of an incompatible lock or latch mode.

I/O WAIT

Elapsed time spent in DB2 waiting for synchronous I/O (QWACAWTI).

Most of the time included in this value is synchronous read time, but under stress conditions it can reflect synchronous write time (when immediate write threshold is reached). During heavy update activity, it also can reflect wait for synchronous log I/O.

Note: Log waits are not included in this value for DB2 6.1 and later.

OTHER READ I/O

Elapsed time spent in DB2 waiting for read I/O, for I/O done by another thread such as sequential prefetch, sequential detection, or list prefetch (QWACAWTR).

This value is the time spent waiting for reads, which are being done under another thread, to complete. These waits may be for sequential prefetch, sequential detection, list prefetch, or for synchronous I/O done while executing a sequential prefetch operation.

Tuning Tip: Time reflected here does not reflect the duration of the I/O but simply the time the thread waited for completion of the I/O activity. If this number is high, parallel processing may assist in reducing the time.

OTHER WRITE I/O

Elapsed time spent in DB2 waiting for write I/O done by another thread (QWACAWTW).

Tuning Tip: Time reflected here should be small, since normal checkpoint and deferred write thresholds protect threads from waiting for asynchronous write time. If this number is large, the buffer pool size or deferred write threshold may need to be adjusted.

UNIT SWITCH EVENTS

In DB2 6.1 and later, the elapsed times spent in DB2 waiting for a synchronous execution unit switch to each of the following DB2 service tasks, which are shown individually:

- Commit, abort, or deallocation processing
- Open/close data set or HSM recall
- SYSLGRNG recording
- Data space manager services (which include define, extend, delete, and reset data set, and VSAM catalog access)
- Other DB2 service tasks

In DB2 5.1 and earlier, these waits are combined into a single value.

ARCH.LOG(QIS)

Elapsed time spent in DB2 waiting for processing of archive log mode (quiesce) commands for the thread (QWACALOG or QWAXALOG in DB2 6.1 and later).

This process externalizes buffers and switches the active log data set. Time reflected here is the time the thread waited, not the time for the command to complete executing.

ARCH.READ(TAPE)

Elapsed time spent in DB2 waiting for archive reads (QWACAWAR or QWAXAWAR in DB2 6.1 and later).

Time spent waiting for an archive tape is either for the completion of the RECOVER utility or for an application program, which has not committed, to read the archive tape in order to complete backout.

Tuning Tip: If this time is large, the application program should commit more often. It may also make sense to increase the size and number of active logs.

DRAIN LOCK

Elapsed time spent in DB2 waiting to acquire drain locks (QWACAWDR or QWAXAWDR in DB2 6.1 and later).

Processes, such as utilities that serialize events to page sets, must acquire a drain lock. This is the time spent waiting for this event.

Tuning Tip: To prevent delays, avoid running utilities during the times that heavy query activity is taking place.

CLAIM RELEASE

Elapsed time spent in DB2 waiting for claim locks to be released after a drain lock was requested (QWACAWCL or QWAXAWCL in DB2 6.1 and later).

Drainers must wait until the SQL claim count reaches zero. This is the time spent waiting for them to clear.

Tuning Tip: To prevent delays, avoid running utilities during the times that heavy query activity is taking place.

PAGELATCH CONT.

Elapsed time spent in DB2 because of page latch contention (QWACAWTP).

Accounting class 3 must be activated to obtain this time.

Tuning Tip: Page latching can be avoided by scheduling applications when there is a lower likelihood of latch contention.

SPAS SERVER TCB

Total elapsed time spent waiting for an available TCB before the Stored Procedure could be scheduled (QWACCAST).

This is the time spent waiting for an available TCB in the Stored Procedures Address Space to become available.

This time should be minimal depending on the workload to the Stored Procedures Address space.

WAITS IN DB2 (GLOBAL)

The elapsed time for the following global DB2 wait events is available only if accounting class 3 is active:

LOCK

Accumulated elapsed wait time caused by an IRLM lock suspension due to global lock contention which requires intersystem communication to resolve (QWACAWTJ).

This time is collected only when data sharing is enabled.

MESSAGE PROC.
Accumulated elapsed wait time caused by suspension for sending messages to other members in the data sharing group (QWACAWTG).

When tables in a data sharing group are CREATed, ALTERed, or DROPped in a data sharing group, intersystem messages must be sent to update the database descriptors (DBDs) in the other members' EDM pools. This is the time spent waiting for this message traffic to occur. This time is collected only when data sharing is enabled.

TOTAL WAITS
Total elapsed time for all DB2 wait events (local and global).

*NOT ACCOUNTED
Elapsed time that could not be accounted for by any DB2 measurements. This is usually some kind of MVS usage, such as DB2 being forced to wait on the MVS dispatcher.

This value is collected only if accounting classes 2 and 3 are activated and is calculated as follows:

In-DB2 Elapsed Time (QWACASC) - In-DB2 CPU (QWACAJST) - CLASS 3 WAIT

Tuning Tip: If this value is high, determine why MVS fails to dispatch DB2.

SQL Statement Execution Counts

The SQL Statement Execution Counts shown in Figure 77 section can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the SQLCOUNTS expand button
- Tabbing to the TOTAL SQL field in the base section and pressing ENTER

This section maintains counts for the individual types of SQL statements executed by this program from start-of-task.

```
- - - - - - - - - - - - SQL STATEMENT EXECUTION COUNTS - - - - - - - - - - - -
SELECT...............0    LOCK TABLE..............0    CREATE............0
INSERT...............0    GRANT/REVOKE............0    DROP..............0
UPDATE...............0    SET CURR.SQLID..........0    ALTER.............0
DELETE...............0    SET HOST VAR. ..........0
                           SET CURR.DEGREE.........0    COMMENT ON........0
PREPARE..............1    SET CURR. RULES.........0    LABEL ON..........0
DESCRIBE.............0    CONNECT TYPE 1..........0
                           CONNECT TYPE 2..........0    CREATE TEMP.......0
OPEN.................1    SET CONNECTION..........0    RENAME TBL........0
FETCH................5    RELEASE.................0
CLOSE................1    CALL....................0
                           ASSOCIATE LOCATOR.......0
                           ALLOCATE CURSOR.........0

** DML...............8    ** DCL..................0    ** DDL............0
** REOPTIMIZED.......0
                   - - - - DYNAMIC SQL CACHE COUNTS - - - -
STMT FOUND...........0    KPDYN-PREP.AVOID........0    STMT DISCRD.......0
STMT NOT FOUND.......0    KEEPDYN_IMPL.PREP.......0    STMT PURGED.......0
```

Figure 77. STRAC SQL Statement Execution Counts (SQLCOUNTS) Section

where

SELECT
Number of SELECT statements executed.

INSERT
Number of INSERT statements executed.

UPDATE
Number of UPDATE statements executed.

DELETE
Number of DELETE statements executed.

PREPARE
Number of PREPARE statements executed.

DESCRIBE
Number of DESCRIBE statements executed.

OPEN
Number of OPEN statements executed.

FETCH
Number of FETCH statements executed.

CLOSE
Number of CLOSE statements executed.

DML
The sum of all Data Manipulation Language (DML) statements executed.

REOPTIMIZED
Indicates the total number of times reoptimization occurred for a query (DB2 Version 5 and later).

LOCK TABLE
Number of LOCK TABLE statements executed.

GRANT/REVOKE
Number of GRANT/REVOKE statements executed.

SET CURR.SQLID
Number of SET CURRENT SQLID statements executed.

SET HOST VAR.
Number of SET HOST-VARIABLE statements executed. (This value does not include the contents of the special register.)

SET CURR.DEGREE
Number of SET CURRENT DEGREE statements executed.

SET CURR. RULES
Number of SET CURRENT RULES statements executed (DB2 Version 4 or above).

CONNECT TYPE 1
Number of CONNECT TYPE 1 statements executed.

CONNECT TYPE 2
Number of CONNECT TYPE 2 statements executed.

SET CONNECTION
Number of SET CONNECTION statements executed.

RELEASE
Number of RELEASE statements executed.

CALL
Number of SQL CALL statements to a Stored Procedure executed (DB2 Version 4 or above).

ASSOCIATE LOCATOR
Number of ASSOCIATE LOCATOR statements executed (DB2 Version 5 and later).

ALLOCATE CURSOR
Number of ALLOCATE CURSOR statements executed (DB2 Version 5 and later).

DCL
Sum of all Data Control Language (DCL) statements executed.

CREATE
Number of CREATE statements executed.

DROP
Number of DROP statements executed.

ALTER
Number of ALTER statements executed.

COMMENT ON
Number of COMMENT ON statements executed.

LABEL ON
Number of LABEL ON statements executed.

CREATE TEMP
Number of CREATE GLOBAL TEMPORARY TABLE statements executed (DB2 Version 5 and later).

RENAME TBL
Number of RENAME TABLE statements executed (DB2 Version 5 and later).

DDL
Sum of all Data Definition Language (DDL) statements executed.

Dynamic SQL Cache Counts:

The following fields are available for DB2 Version 5 and later:

STMT FOUND
Indicates the number of times a prepare request was satisfied by making a copy from the prepared statement cache.

STMT NOT FOUND
Indicates the number of times a prepare request was received but a matching statement was not found in the prepared statement cache.

KPDYN-PREP.AVOID
Indicates the number of times that a prepare was avoided when these conditions existed:

- The KEEPDYNAMIC(YES) option was used along with prepare statement caching
- DB2 still had the application process copy of the executable version of the prepared statement

KEEPDYN_IMPL.PREP
Indicates the number of times that an implicit prepare was performed when these conditions existed:

- The KEEPDYNAMIC(YES) option was used
- An open, execute, or describe of a dynamic statement occurred after a commit
- DB2 no longer had a valid copy of the executable version of the prepared statement

STMT DISCRD
Indicates the number of times that a prepared statement was discarded due to the MAXKEEPD system limit being exceeded.

STMT PURGED
Indicates the number of times that a prepared statement was purged from the cache because a drop, alter, or revoke statement was issued on a dependent object.

Buffer Pool Activity

The Buffer Pool Activity section shown in Figure 78 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the BPOOL expand button
- Tabbing to the GETPAGES field in the base section and pressing ENTER

```
- - - - - - - - - - - - - - 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
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
NBR UNREG PAGES....         1         -         1
NBR EXPLICIT XI....         0         -         0
NBR WRITES 2ND BP..         0         -         0
```

Figure 78. STRAC Buffer Pool Activity (BPOOL) Section

where

GETPAGES
: Number of GETPAGE I/O requests.

SYNC READS
: Number of synchronous read I/Os.

GETPAGES/READIO
Number of GETPAGEs per synchronous READIO requests.

This ratio is only meaningful for random activity. If there is substantial asynchronous prefetch activity, review the value in ASYNC PAGES READ. You may want to calculate a total ratio including this value for a particular buffer pool. The VP HIT RATIO for the thread in the base section of the display includes both synchronous and asynchronous I/Os.

COND. GP FAILURES
Number of unsuccessful GETPAGE operations due to conditional GETPAGE requests.

SEQ. PREFETCH REQS
Number of sequential prefetch requests.

LIST PREFETCH REQS
Number of list prefetch requests.

DYNAMIC PREFETCHES
Number of dynamic prefetch requests.

ASYNC PAGES READ
Number of asynchronous pages read.

PAGES/PREFETCH REQ
The ratio of GETPAGEs per prefetch request.

PAGE UPDATES
Number of System Page Update requests done by this task. This counter is incremented by one each time a row in a page in the buffer pool is updated. It is also incremented for work file pages.

IMMEDIATE WRITES
Number of immediate (synchronous) write I/Os.

HP SYNC READS
Number of successful hiperpool synchronous reads.

HP SYNC READ FAILS
Number of unsuccessful hiperpool synchronous reads.

HP ASYNC PAGES READ
Number of hiperpool asynchronous pages read.

HP WRITES
Number of successful hiperpool writes.

HP WRITE FAILURES
Number of unsuccessful hiperpool writes.

Global Buffer Pool (Data Sharing Only):

The following fields are available if data sharing is active and has been used by this transaction.

CF READS (BUFFER CROSS INVALIDATION)
Coupling Facility READ requests required because the buffer was marked INVALID. The statistics are given for each of these circumstances:

-DATA RETURNED
Data is returned from the group buffer pool.

-R/W INTEREST
Data is not returned from the group buffer pool and a directory entry is created if it does not already exist, because another DB2 in the group has READ/WRITE interest in the page set or partition.

-NO R/W INTEREST
Data is not returned from the group buffer pool and no directory entry is created for this page, because no other DB2 in the group has READ/WRITE interest in the page set or partition.

CF READS (DATA NOT IN BUFFER POOL)
Coupling Facility READs necessary because the requested page was not found in the buffer pool. The statistics are given for each of these circumstances:

-DATA RETURNED
Data is returned from the Coupling Facility.

-R/W INTEREST
Data is not returned from the Coupling Facility and a directory entry is created if it does not already exist, because another DB2 in the group has READ/WRITE interest in the page set or partition.

-NO R/W INTEREST
Data is not returned from the Coupling Facility and no directory entry is created for this page, because no other DB2 in the group has READ/WRITE interest in the page set or partition.

CF-WRITE (CHGD PGS)
Number of changed pages written to the group buffer pool.

CF-WRITE (CLEAN PG)
Number of clean pages written to the group buffer pool. (DB2 writes clean pages for page sets and partitions defined with GBPCACHE ALL.)

NBR UNREG PAGES
Number of coupling facility requests to unregister a page. (DB2 version 5 and later)

NBR EXPLICIT XI
Number of explicit cross-invalidations. (DB2 version 6 and later)

NBR WRITES 2ND BP
Number of coupling facility requests to write changed pages to the secondary group buffer pool for duplexing. (DB2 version 6 and later)

Lock Activity

The Lock Activity section shown in Figure 79 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the LOCKS expand button
- Tabbing to the UPDATES/COMMIT field in the base section and pressing ENTER.

```
- - - - - - - - - - - - - - LOCK ACTIVITY  - - - - - - - - - - - - -  - -
TIMEOUT.........................0     MAX PAGE/ROW LOCKS...............1
DEADLOCK........................0
                                      LOCK REQUEST.....................6
SUSPEND-LOCK....................0     UNLOCK REQUEST...................9
SUSPEND-LATCH...................0     QUERY REQUEST....................0
SUSPEND-OTHER...................0     CHANGE REQUEST...................8
                                      OTHER REQUEST....................0
ESCALATION(SHR).................0
ESCALATION(EXCL)................0

DRAIN REQUEST...................0     DRAIN FAILURE....................0
CLAIM REQUEST...................2     CLAIM FAILURE....................0
                     - - - - GLOBAL LOCK ACTIVITY - - - -
LOCK REQUEST....................1     XES LOCK REQUEST.................3
LOCK CHANGE REQUEST.............0     XES CHANGE REQUEST...............0
UNLOCK REQUEST..................0     XES UNLOCK REQUEST............. 1

IRLM GLOBAL SUSPEND.............0     XES GLOBAL SUSPEND...............0
FALSE CONTENTION................0

INCOMPATIBLE RETAINED...........0     NOTIFY MESSAGES SENT............0
```

Figure 79. STRAC Lock Activity (LOCKS) Section

where

TIMEOUT
: Timeout counts.

DEADLOCK
: Deadlock counts.

SUSPEND-LOCK
: Number of suspends because of lock conflicts.

SUSPEND-LATCH
: Number of suspends because of latch conflicts.

SUSPEND-OTHER
: Number of suspends because of other conflicts.

ESCALATION(SHR)
: Number of lock escalations to shared mode.

ESCALATION(EXCL)
: Number of lock escalations to exclusive mode.

DRAIN REQUEST
Number of drain requests

CLAIM REQUEST
Number of claim requests

MAX PAGE/ROW LOCKS
Maximum number of page or row locks held.

LOCK REQUEST
Lock request count.

UNLOCK REQUEST
Unlock request count.

QUERY REQUEST
Query request count.

CHANGE REQUEST
Change request count.

OTHER REQUEST
Other IRLM request count.

DRAIN FAILURE
Number of drain requests that were unsuccessful.

CLAIM FAILURE
Number of claim requests that were unsuccessful.

Global Lock Activity (Data sharing Only):

The following fields are available if the system is part of a data sharing group:

LOCK REQUEST
Number of LOCK requests for physical locks (P-locks).

LOCK CHANGE REQUEST
Number of CHANGE requests for physical locks (P-locks).

UNLOCK REQUEST
Number of UNLOCK requests for physical locks (P-locks).

IRLM GLOBAL SUSPEND
Number of suspensions because of IRLM global resource contention. (IRLM lock states were in conflict.)

FALSE CONTENTION
Number of suspensions caused by false contentions.

This occurs when different resource names hash to the same entry in the Coupling Facility lock table. This causes MVS XES to detect contention on the hash class; however, when MVS XES determines that there is no real conflict on the resource, the contention is called *false*.

INCOMPATIBLE RETAINED
Number of global lock or change requests denied because of an incompatible retained lock.

XES LOCK REQUEST
Number of LOCK requests (both L-locks and P-locks) propagated to MVS XES synchronously (under the user's execution unit).

XES CHANGE REQUEST
Number of CHANGE requests (both L-locks and P-locks) propagated to MVS XES synchronously (under the user's execution unit).

XES UNLOCK REQUEST
Number of resources propagated to MVS XES synchronously (under the user's execution unit) from UNLOCK requests (both L-locks and P-locks).

XES GLOBAL SUSPEND
Number of suspensions because of MVS XES global resource contention. (MVS XES lock states were in conflict but IRLM lock states were not.)

NOTIFY MESSAGES SENT
Number of NOTIFY messages sent to other members of this data sharing group.

Parallelism

The Parallelism section shown in Figure 80 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the PRLL expand button
- Tabbing to the PRLL=nn field in the base section and pressing ENTER

Note: With DB2 3.1, the section header reads: I/O PARALLELISM.

In DB2 5.1 and later, this also includes utility parallel tasks.

```
- - - - - - - - - - - - - - - - PARALLELISM  - - - - - - - - - - - - - - - - - -
MAXIMUM DEGREE.................2        FALLBACK - NO BUFFER............1
GROUPS EXECUTED................3        FALLBACK - NO ESA SORT..........0
-- PLANNED DEGREE..............1        FALLBACK - AMBIG. CURSOR........1
-- REDUCED DEGREE..............2

PARALLEL TASKS.................3        FALLBACK - NO ENCLAVE...........0
CORRELATION TOKEN.....X'00000000'        FALLBACK - RLF LIMITED.........NO
                        - - - - SYSPLEX - - - -
COORDN/ASSISTANT..........COORDN        FALLBACK - NO COORDN............1
PRLL COORDN NAME................        FALLBACK - ISOL RR/RS...........2
MAXIMUM MEMBERS................2        REDUCED  - VP SHORTAGE..........0
INTENDED PRLL GROUPS...........0
QUERY REFORMULATED (ENV)........0        QUERY REFORMULATED (BP).........0
```

Figure 80. STRAC Parallelism (PRLL) Section

where

MAXIMUM DEGREE
: Maximum degree of parallelism achieved among the parallel groups. This field indicates the extent to which queries were processed in parallel.

GROUPS EXECUTED
: Total number of parallel groups executed.

PLANNED DEGREE
: Total number of parallel groups that have a planned degree greater than one and were successfully executed at the planned degree.

REDUCED DEGREE
: Total number of parallel groups that have a planned degree greater than one, but were processed at a lesser parallel degree because of a storage shortage or contention on the buffer pool.

  If this field is not zero, consider increasing the size of the current buffer pool or reassigning the table space to another buffer pool.

FALLBACK - NO BUFFER
: Total number of parallel groups that have a planned degree greater than one, but fell back to sequential mode because of storage shortage or contention on the buffer pool.

FALLBACK - NO ESA SORT
Total number of parallel groups that fell back to sequential mode due to lack of ESA sort support.

FALLBACK - AMBIG. CURSOR
Total number of parallel groups that fell back to sequential mode because the cursor is ambiguous (may be used for UPDATE or DELETE).

PARALLEL TASKS
Total number of subtasks that were created to support parallel tasks for this query or utility. (Available only if accounting class 1 is active.)

CORRELATION TOKEN
For subtasks created to support a parallel query or utility, this field contains the correlation token of the originating task. This token is equal to the address of the Agent Control Element (ACE) of the originating task. This token can be used to correlate parallel subtask records with the records of the originating task. This field is set to zeros for the originating task. (Available only if accounting class 1 is active.)

FALLBACK - NO ENCLAVE
Total number of parallel groups executed in sequential mode because MVS/ESA enclave services were not available.

FALLBACK - RLF LIMITED
YES|NO. Indicates whether query parallelism is disabled by the Resource Limit Facility (RLF) for at least one dynamic select statement in this thread.

Sysplex Parallelism:

The following fields are available for DB2 Version 5 and later:

COORDN/ASSISTANT
Indicates that a query is being processed across one DB2 (coordinator) or more than one DB2 (assistant).

PRLL COORDN NAME
Parallel coordinator name of the DB2 member that received the query for processing in a sysplex.

MAXIMUM MEMBERS
Indicates the maximum number of DB2 members that participated in the processing of a query.

INTENDED PRLL GROUPS
Indicates the total number of parallel groups that were intended to run across the data sharing group. This value is only incremented by the coordinator member at run time.

QUERY REFORMULATED (ENV)
Total number of parallel groups for which DB2 reformulated the parallel portion of the access path because the sysplex configuration at run time was different from the sysplex configuration at bind time.

This counter is incremented only by the parallelism coordinator at run time.

(DB2 6.1 and later)

FALLBACK — NO COORDN
Indicates the total number of parallel groups executed on a single DB2 (no sysplex).

FALLBACK — ISOL RR/RS
Indicates the total number of parallel groups executed on a single DB2 member due to repeatable-read or read-stability isolation (no sysplex).

REDUCED — VP SHORTAGE
Indicates the number of times that the parallelism coordinator had to bypass a DB2 when distributing tasks because there was not enough buffer pool storage on one or more DB2 members.

QUERY REFORMULATED (BP)
Total number of parallel groups for which DB2 reformulated the parallel portion of the access path because there was not enough buffer pool resource.

This counter is incremented only by the parallelism coordinator at run time.

(DB2 6.1 and later)

Routines

The Routines section shown in Figure 81 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the RTN expand button

Note: Each part of this section is available only if the user connection has issued stored procedure calls or user-defined functions or has been invoked by a trigger.

```
- - - - - - - - - - - - - - -STORED PROCEDURES- - - - - - - - - - - - - - -
SPAS CPU TIME...............200 us        SQL CALLS.......................9
SPAS IN-DB2 TIME............200 us        CALLS ABENDED...................0
SPAS WAITING ON TCB.........600 ms        CALLS REJECTED..................0
                                          CALLS TIMED OUT.................0
Total elapsed time..........200 us        WAITS ON TCB....................6
Elapsed time in SQL.........200 us
                                          SQL nest level..................5

---and/or--

- - - - - - - - - - - - USER-DEFINED FUNCTIONS- - - - - - - - - - - - - - -
UDF CPU TIME................200 us        UDFs executed...................9
UDF IN-DB2 TIME.............200 us        UDFs ABENDED....................0
UDF WAITING ON TCB..........600 ms        UDFs REJECTED...................0
                                          UDFs TIMED OUT..................0
Total elapsed time..........200 us
Elapsed time (SQL)..........200 us        SQL STMTS executed.............15
                                          SQL nest level..................5

---and/or--

- - - - - - - - - - - - - - - TRIGGERS  - - - - - - - - - - - - - - - - - -
-- non-enclave times --                   Stmt TRG activated..............9
CPU Time....................200 us        Row TRG activated...............0
Elapsed time................600 ms        Trigger SQL errors..............0
-- enclave times --
CPU time....................200 us        CPU time (before enclave)...200 us
Elapsed time................200 us
                                          SQL nest level..................5
```

Figure 81. STRAC Routines (RTN) Section

where

Stored Procedures:

SPAS CPU TIME
Accumulated TCB time spent processing SQL CALL statements in the DB2 stored procedures address space (available only if accounting class 1 is active).

SPAS IN-DB2 TIME
Accumulated TCB time spent in DB2 processing SQL statements issued by stored procedures (available only if accounting class 2 is active).

SPAS WAITING ON TCB
Total elapsed time spent waiting for an available TCB before the stored procedure could be scheduled (available only if accounting class 3 is active).

SQL CALLS
Number of SQL CALL statements executed.

CALLS ABENDED
Number of times a stored procedure terminated abnormally.

CALLS REJECTED
Number of times an SQL CALL statement was rejected because the procedure was in the 'STOP ACTION(REJECT)' state.

CALLS TIMED OUT
Number of times an SQL CALL statement timed out while waiting to be scheduled.

WAITS ON TCB
Number of times an SQL CALL statement had to wait for an available TCB before the stored procedure could be scheduled (available only if accounting class 3 is active).

Total elapsed time
Total elapsed time spent in stored procedures, including time spent executing SQL (DB2 6.1 and above).

Elapsed time in SQL
Total elapsed time spent executing SQL in stored procedures (DB2 6.1 and above).

SQL nest level
Maximum level of nested SQL cascading due to triggers, user-defined functions, and stored procedures (DB2 6.1 and above).

Note: This field is displayed only if SQL nesting has taken place.

User-Defined Functions:

The following fields are available for DB2 version 6 and above if user-defined functions have been used:

UDF CPU TIME
: Accumulated TCB time used to satisfy user-defined function requests processed in a DB2 stored procedures address space or WLM-established address space.

  This time is collected with accounting class 1 active.

UDF IN-DB2 TIME
: Accumulated TCB time in DB2 spent processing SQL statements issued by user-defined functions.

  This is the time not included in UDF CPU TIME.

  This number is calculated only if accounting class 2 is active.

UDF WAITING ON TCB
: Total elapsed time spent waiting for an available TCB before the user-defined function could be scheduled.

  This value is calculated only if accounting trace class 3 is active.

Total elapsed time
: Total elapsed time spent in user-defined functions, including time spent executing SQL.

  This value is calculated only if accounting trace class 1 is active.

Elapsed time (SQL)
: Total elapsed time spent for user-defined functions to execute SQL.

  This value is calculated only if accounting trace class 3 is active.

UDFs executed
: Number of user-defined functions executed.

UDFs ABENDED
: Number of times a user-defined function abended.

UDFs REJECTED
: Number of times a user-defined function was rejected.

UDFs TIMED OUT
: Number of times a user-defined function timed out while waiting to be scheduled.

SQL STMTS executed
: Number of SQL entry/exit events performed by user-defined functions.

  This value is calculated only if accounting trace class 2 is active.

SQL nest level
: Maximum level of nested SQL cascading due to triggers, user-defined functions, and stored procedures.

  Note: This field is displayed only if SQL nesting has taken place.

Triggers:

The following fields are available for DB2 version 6 and above if a trigger was invoked:

CPU Time (non-enclave)
: Accumulated TCB time used while executing under the control of triggers.

Elapsed time (non-enclave)
: Accumulated elapsed time used while executing under the control of triggers.

CPU time (enclave)
: Accumulated TCB time used for executing triggers under an enclave.

Elapsed time (enclave)
: Accumulated elapsed time used for executing triggers under an enclave.

Stmt TRG activated
: Number of times a statement trigger is activated.

Row TRG activated
: Number of times a row trigger is activated.

Trigger SQL errors
: Number of times an SQL error occurred during execution of a triggered action.

CPU time (before enclave)
: Accumulated TCB time used before the enclave is created.

SQL nest level
: Maximum level of nested SQL cascading due to triggers, user-defined functions, and stored procedures.

Note: This field is displayed only if SQL nesting has taken place.

DDF Summary

The DDF Summary section shown in Figure 82 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the DDF expand button

This section is available only if there is DDF activity for the displayed thread.

Note: There can be multiple locations per thread and also multiple conversations per location.

```
- - - - - - - - - - - - - - - - - - DDF SUMMARY - - - - - - - - - - - - - - - - -
CURRENT LOCATION: DB2D (DBAT SERVER/ALLIED DIST REQUESTOR)
   WORKSTATION USER ID...bolbpl2          WORKSTATION NAME..
   WORKSTATION TRANSACTION ID...db2bp.exe

   MVS WLM SERVICE CLASS NAME......DB2JWLM

REMOTE            FROM         TO
  LOCATION:  DB1D                        SQL STATEMENTS           0          1
  REMOTE PROD-ID:    DB2PROD             ROWS                   250          0
  DB AGENT CPU...........505 ms          BYTES                63,237      1,237
  ELAPSED LOCAL.........00:02:19         TRANSACTIONS             0          1
  ELAPSED REMOTE........00:02:18         MESSAGES                 5          6
  CONVERSATIONS QUEUED......0            CONVERSATIONS            0          1
  BLOCK FETCH %............21%          COMMITS                  0          1
  MAX CONVERSATIONS.........1            ABORTS                   0          0
  UNSUCCESSFUL CONV'S.......0            BLOCK FETCH             20          0

TWO-PHASE COMMIT (2PH):                  2PH PREPARE              0          0
                                         2PH LAST AGENT           0          0
REMOTE COORDINATOR                       2PH COMMITS              0          0
------------------                       2PH BACKOUTS             0          0
-- INDOUBT................1             2PH FORGET RSP           0          0
-- COMMIT.................3             2PH REQ COMMIT           0          0
-- ROLLBACK...............0             2PH BACKOUT RSP          0          0
```

Figure 82. STRAC DDF Summary (DDF) Section

where

CURRENT LOCATION:
Current 16-character location name.

WORKSTATION USER ID:
End user's workstation user ID. This may be different from the AUTHID used to connect to DB2.

This field contains blanks if the client did not supply this information.

Note: This field is not displayed if the user is not a workstation client.

WORKSTATION TRANSACTION ID:
Transaction or application name that the end user is running. This identifies the application that is currently running (e.g. ORDER_UPDATE), rather than the product that is being used to run the application (e.g. CICS/6000).

This field contains blanks if the client did not supply this information.

Note: This field is not displayed if the user is not a workstation client.

WORKSTATION NAME:
Name of the end user's workstation.

This field contains blanks if the client did not supply this information.

Note: This field is not displayed if the user is not a workstation client.

MVS WLM SERVICE CLASS NAME
The MVS workload manager (WLM) service class name. This field is used only for database access threads on MVS 5.2 or later. (DB2 5.1 and later)

Note: The WLM service class determines the MVS WLM priority of the work performed by the database access thread.

REMOTE LOCATION:
Location name of the remote site with which the information is associated.

REMOTE PROD-ID:
Product ID of the remote location.

DB AGENT CPU
Database access agent CPU time at the remote site.

ELAPSED LOCAL
Elapsed time at the local site spent processing for the remote site.

ELAPSED REMOTE
Elapsed database access agent time at the remote site (accounting only). This data is not supported for application-directed access and is misleading if both system-directed access and application-directed access flows are used.

CONVERSATIONS QUEUED
Number of conversation requests queued by DDF waiting for allocation.

BLOCK FETCH%
Number of rows transmitted by block fetch as a percentage of the total rows transmitted.

MAX CONVERSATIONS
Maximum number of conversations open at any one time.

UNSUCCESSFUL CONV'S
Number of unsuccessful conversations.

SQL STATEMENTS
Number of SQL statements sent to and from remote site.

ROWS
Number of data rows sent to and from remote site.

BYTES
Number of bytes of data sent to and from remote site.

TRANSACTIONS
Number of transactions sent to and from remote site.

MESSAGES
Number of messages sent to and from remote site.

CONVERSATIONS
Number of conversations sent to and from remote site.

COMMITS
Number of commit requests sent to and from remote site.

ABORTS
Number of abort requests sent to and from remote site.

BLOCK FETCH
Number of blocks transmitted using block fetch.

TWO-PHASE COMMIT (2PH):

The following statistics are provided for two-phase commit operations:

REMOTE COORDINATOR

The following statistics are provided if there is a remote coordinator:

INDOUBT

Number of threads that went indoubt with the remote location as coordinator.

COMMIT

Number of commit requests sent to and from the remote site with the remote location as coordinator.

ROLLBACK

Number of rollback operations performed with the remote location as coordinator.

2PH PREPARE

Number of prepare requests sent to the participant and received from the coordinator for two-phase commit operations.

2PH LAST AGENT

Number of last agent requests sent to the coordinator and received from the initiator for two-phase commit operations.

2PH COMMITS

Number of commit requests sent to the participant and received from the coordinator for two-phase commit operations.

2PH BACKOUTS

Number of backout requests sent to the participant and received from the coordinator for two-phase commit operations.

2PH FORGET RSP

Number of forget responses sent to the coordinator and received from the participant for two-phase commit operations.

2PH REQ COMMIT

Number of request commit responses sent to the coordinator and received from the participant for two-phase commit operations.

2PH BACKOUT RSP

Number of backout responses sent to the coordinator and received from the participant for two-phase commit operations.

Package/DBRM Overview

The Package/DBRM Overview section shown in Figure 83 can be viewed by any of these methods:

- Scrolling down from the initial STRAC display
- Selecting the PKG expand button

Note: This section is available only when accounting class 7 is active.

```
- - - - - - - - PACKAGE / DBRM OVERVIEW (ACCTG CLASSES 7,8 ONLY) - - - - - - -

PACKAGE/                                          %TOTAL
DBRM      NO. SQL  CPU TIME  WAIT TIM  ELAPSED   ELAP.
-------- --------  --------  --------  -------- ------ 0 ...25...50...75..100
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 83. STRAC Package/DBRM Overview (PKG) Section

where

PACKAGE/DBRM
: Package or DBRM name. An asterisk (*) after this column indicates this package or DBRM was loaded by a stored procedure.

NO. SQL
: Number of SQL statements issued in this package or DBRM.

Note: This field may not equal the total of all SQL statements in the SQL Statement Execution Counts section (DSNDQXST). This is because that section does not include all SQL statements, such as COMMIT or ROLLBACK.

CPU TIME
: Total TCB CPU time for all executions of this package or DBRM.

WAIT TIM
: Total elapsed wait time in DB2 while executing in this package or DBRM.

ELAPSED
: Total elapsed time in DB2 spent processing all executions of this package or DBRM.

% TOTAL ELAP.
: Percent of total elapsed time spent in DB2 for all executions of this package or DBRM. This value is also displayed graphically.

Expand: From the Package/DBRM Overview section of STRAC, you can expand to:

Package/DBRM Pop-up
: Place the cursor on a line of data for a package or DBRM in the Package/DBRM Overview section of STRAC and press ENTER to view a pop-up display of detail statistics for that package or DBRM. See "Package/DBRM Pop-up" on page 239 for a description of this pop-up.

Package/DBRM Pop-up

Place the cursor on a line of data for a package or DBRM in the Package/DBRM Overview section of STRAC and press ENTER to view the following pop-up display. It summarizes all statistics from the executions of one package or DBRM.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> STRAC           INPUT    11:12:21  INTVL=> 3  LOG=> N  TGT==> DB2E
PARM ==> THRDHIST,SEQ=6                                          SCROLL=> CSR

- - - - - - - - PACKAGE / DBRM SECTION (ACCTG CLASSES 7,8 ONLY) - - - - - - -

NAME:            DB2E.BOLLA1.ACUT0300
TYPE:            PACKAGE                LOCATION:           DB2E
PROGRAM:         ACUT0300               COLLECTION ID:      BOLLA1
    LOADED BY A STORED PROCEDURE        CONSISTENCY TOKEN:  152B76830BA4B826

SQL STATEMENTS -  TOTAL:        8
DB2 ENTRY/EXIT -  TOTAL:       29

TIMINGS            #EVENTS AVG/EVENT   IN-DB2  %TOTAL
-----------------  ------- --------- -------- ------- 0 ...25...50...75..100
ELAPSED TIME                           139 ms    0.15 | <                    |
TCB CPU TIME                         7,831 us    0.00 |                      |
TOTAL WAITS:            14 8,250 us    116 ms    0.13 | <                    |
  LOCK/LATCH             0     0 us      0 us    0.00 |                      |
  I/O WAIT               8 4,451 us     36 ms    0.04 | <                    |
  OTHER READ             0     0 us      0 us    0.00 |                      |
  OTHER WRITE            0     0 us      0 us    0.00 |                      |
  UNIT SWITCH            6    13 ms     80 ms    0.09 | <                    |
  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 |                      |
  GLOBAL LOCK            0     0 us      0 us    0.00 |                      |
  DATA SHR MSGS.         0     0 us      0 us    0.00 |                      |
  UDF Schedule           0     0 us      0 us    0.00 |                      |
*NOT ACCOUNTED                         307 us    0.00 |                      |
```

Figure 84. STRAC Package/DBRM Pop-up Display

where

NAME
: Package or DBRM name.

TYPE
: Specifies either PACKAGE or DBRM.

PROGRAM
: Program name.

LOADED BY A STORED PROCEDURE
: Displayed if the selected package or DBRM was loaded in response to a stored procedure.

LOCATION
: Location name where package or DBRM was executed.

COLLECTION ID
Package collection ID.

CONSISTENCY TOKEN
Token to identify this package or DBRM.

SQL STATEMENTS
Total number of SQL statements issued in this package or DBRM.

DB2 ENTRY/EXIT
Total number of entries to and exits from DB2 for this package or DBRM.

The following area displays the elapsed time in DB2, number of events, average elapsed time per event, and percent of total elapsed time for each of the following categories while this package or DBRM was executing. The percent of total elapsed time is also displayed graphically for each of these categories.

ELAPSED TIME
Total elapsed time spent processing all executions of this package or DBRM.

TCB CPU TIME
Total TCB CPU time for all executions of this package or DBRM.

TOTAL WAITS:
Total time spent waiting for each of the following resources in DB2 while executing this package or DBRM.

The elapsed time for the following DB2 wait events is available only if accounting class 8 is active:

LOCK/LATCH
Elapsed time spent in DB2 waiting for a lock or latch.

I/O WAIT
Elapsed time spent in DB2 waiting for synchronous I/O. This includes synchronous reads or writes as well as log write commit I/O.

OTHER READ
Elapsed time spent in DB2 waiting for read I/O, for I/O done by another thread such as sequential prefetch, sequential detection, or list prefetch.

OTHER WRITE
Elapsed time spent in DB2 waiting for write I/O done by another thread.

UNIT SWITCH
Elapsed time spent in DB2 waiting for work done under another execution unit, such as

- Commit, abort, or deallocation processing

In DB2 5.1 and below, the following waits are also included in this value:

- HSM recall
- Open/close data set
- SYSLGRNG update
- Data space manager services
- Define, extend, or delete data set

ARCH.LOG(QIS)
: Elapsed time spent in DB2 waiting for processing of archive log mode (quiesce) commands for the thread.

ARCH.READ(TAPE)
: Elapsed time spent in DB2 waiting for archive reads.

DRAIN LOCK
: Elapsed time spent in DB2 waiting to acquire a drain lock.

CLAIM RELEASE
: Elapsed time spent in DB2 waiting for claim holders to release the object after the drain lock is acquired.

PAGELATCH CONT.
: Elapsed time spent in DB2 because of page latch contention.

SPAS SERVER TCB
: Elapsed time spent waiting for an available TCB before the stored procedure could be scheduled.

GLOBAL LOCK
: Accumulated elapsed wait time caused by suspension of an IRLM lock request due to global lock contention.

DATA SHR MSGS.
: Accumulated elapsed wait time caused by suspension for sending messages to other members in the data sharing group.

UDF Schedule
: Elapsed time spent waiting for scheduled user-defined functions (DB2 version 6 and later only).

*NOT ACCOUNTED
: Elapsed time that could not be accounted for as In-DB2 CPU or In-DB2 Waits. This usually is some kind of MVS usage, such as dispatcher waits.

Detail Trace Summary Sections

The following sections format data summarized from detail trace events and are available only if these events are traced.

SQL Summary

The summaries shown in Figure 85 on page 243 can be viewed by scrolling down from the initial STRAC display or by selecting the SQL expand button.

Note: This section is not shown for a summary trace. The heading line is shown with no data if the display is for a detail trace that does not include SQL events.

When only the SQL section is displayed, the default order of presentation is by program, and the display can be sorted by any column (see “STRAC SQL Summary Sort” in this section).

Cursor selection in this section displays the following:

- SQL Statement Pop-up (statement line)
- DTRAC qualified by program name (PGM line)

See “STRAC SQL Summary Expand” in this section.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC  INPUT    14:27:16  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,SQL,SORT=PGM    ROW 1 OF 18 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, 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+
------- -----  -----  -------- -----  -------- -----  ----  ---- ---- ----
SELECT   3228      1     25 ms   2.7  1,930 us   0.3     0     2    1    0
SELECT   3347      2     11 ms   2.4  4,047 us   1.5     0    11    4    0
OPEN     3565      6    172 us   0.1    170 us   0.2     0     0    0    0
FETCH    3578     11  1,986 us   2.3    782 us   1.6     0    19    3    0
CLOSE    3664      6    192 us   0.1    149 us   0.2     0     0    0    0
SELECT   3671      6  1,776 us   1.1  1,091 us   1.2     0    12    0    0
SELECT   3283      1  2,643 us   0.3  1,888 us   0.3     0     1    0    0
OPEN     3299      1    109 us   0.0    107 us   0.0     0     0    0    0
FETCH    3313      2    907 us   0.2    855 us   0.3     0     1    0    0
CLOSE    3334      1    142 us   0.0    141 us   0.0     0     0    0    0
OPEN     3456      1    693 ms  74.1    414 ms  74.9     8    23 1187    2
FETCH    3468      5    440 us   0.2    322 us   0.3     0     0    0    6
SELECT   4803      1  1,269 us   0.1  1,121 us   0.2     0     2    0    0
PGM: P025D100     44            83.9            81.0     8    71 1195    8

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

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

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

Figure 85. STRAC SQL Summary Section

Expand: From the SQL Summary section of STRAC you can expand to these two displays:

SQL Statement Pop-up

Place the cursor on a line of data for an SQL statement in the SQL Summary section of STRAC and press ENTER to view a pop-up display that summarizes all statistics from the individual occurrences of that statement. See "SQL Statement Pop-up" on page 246 for a description of this pop-up.

DTRAC PGM=

Place the cursor on the PGM line in the SQL Summary section of STRAC and press ENTER to request a DTRAC display of only those events from a given program.

The statements displayed by DTRAC are qualified by the program name specified by the PGM field. Only those SQL statements issued by that originating program and all events subordinate to the selected SQL statements are shown. For more information about the DTRAC display, see "DTRAC—Detail Trace Entry" on page 255.

Sort: A qualified SQL display can be sorted by any column. Some SQL columns can be sorted by individual SQL statement or by program summary.

To request a sort of the qualified STRAC SQL Summary display, type the following in the STRAC PARM field:

```
,SORT=value
```

where value can be

PGM	Program name - alphanumeric ascending sort
ST	Statement type - alphabetic ascending sort

A sort request by individual statement displays the program associated with that statement below the data line whenever the program name changes.

CO	Count
AE	Average elapsed
%E	Percent elapsed
AC	Average CPU
%C	Percent CPU
SR	Sort records
PI	Pages scanned, index
PD	Pages scanned, data
PW	Pages scanned, work
PR	Pages scanned for referential integrity

A program summary sort keeps all of the SQL statements for a program together in one group, arranging the group displays by the sorted column.

COP	Count by program
%EP	Percent elapsed by program
%CP	Percent CPU by program
SRP	Sort records by program
PIP	Pages scanned, index, by program
PDP	Pages scanned, data, by program
PWP	Pages scanned, work, by program
PRP	Pages scanned for referential integrity, by program

All numeric sorts are in descending order.

Description: These statistical summaries for a single transaction are as follows:

STMT TYPE
SQL statement type.

STMT
DB2 precompiler statement number.

COUNT
Number of times the statement executed.

AVG. ELAPSED
Average elapsed time per statement.

% ELAP
Percentage of total elapsed time for the plan that this statement consumed.

AVG. CPU
Average CPU time per statement.

% CPU
Percentage of total CPU time for the plan this statement consumed.

SORT RECS
Number of records sorted by DB2 to satisfy the request.

PAGES SCANNED
Number of pages scanned in index, data, and work (DSNDB07) page sets and in temporary tables and transition tables.

PGM:
Name of the program. A series of periods (.) in this field indicates an unidentifiable program name. Other columns in this line show program or program percentage totals.

TOTALS
Totals for this plan execution.

Note: These totals may differ from the totals in the SQLCOUNTS section (Figure 77 on page 216) because DESCRIBE statements are not included in this section.

SQL Statement Pop-up

Place the cursor on a line of data for an SQL statement in the SQL Summary section of STRAC and press ENTER to view the following pop-up display. It summarizes all statistics from the individual occurrences of one statement, providing quick access to row processing information without the need to scan through detail events in DTRAC.

```
BMC SOFTWARE---------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> STRAC           INPUT    10:22:51  INTVL=> 3  LOG=> N  TGT==> DB2C
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                             ----------------------------
                                                  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
LOB- PAGES SCANNED                                    0         0         0
   - PAGES UPDATED                                    0         0         0
```

Figure 86. STRAC SQL Statement Pop-up Display

Expand: From the STRAC SQL Statement pop-up you can expand to:

SQLTEXT(EXPLAIN)
: A display of the complete statement text, if the statement is static SQL and can be explained, and if RxD2 is installed.

From this panel you can choose to:

- EXPLAIN the text and access the RxD2 EXPLAIN display for the statement

  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.

- 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.

- 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: Access is possible to remote DB2 subsystems only if connected with DDF to a local DB2 (in the same MVS system as the TS).

Description: A maximum of five columns can be displayed showing the individual statistics for index (INDEX), sequential data (SEQ-DATA), work tables (SEQ-WORK), temporary tables (SEQ-TEMP), and transition tables (SEQ-TRAN). Only columns for the types of scan used are displayed.

Each row describes the type of processing that DB2 performed to satisfy the SQL statement by the following:

STATEMENT:
Number and type of statement.

PLAN:
Plan name.

PROGRAM:
Name of the program that issued the SQL statement. If this field contains a series of periods (.), the program name detected is unidentifiable.

LOCATION:
Name of the location associated with the thread. This field can be blank. If it is not blank, it is displayed as shown in Figure 86 on page 246.

PACKAGE:
Package collection ID. This field can be blank. If it is not blank, it is displayed as shown in Figure 86 on page 246.

NUMBER OF EXECUTIONS:
Number of times this statement is executed.

ELAPSED: AVERAGE
Average elapsed time per statement.

ELAPSED: TOTAL
Total elapsed time per statement.

CPU: AVERAGE
Average CPU time per statement.

CPU: TOTAL
Total CPU time per statement.

ROWS PROCESSED ALL TYPES
Number of rows processed in the table space of all types. This figure is meaningful when a table space contains more than one table. Rows for other tables must also be accessed to determine whether they are required to satisfy the request.

ROWS PROCESSED CORRECT TYPE
Number of rows processed that were of the correct type. For a single table space, this should equal ROWS PROCESSED ALL TYPES.

ROWS QUALIFIED BY DATA MANAGER
Number of rows that qualified for the request by DB2's Data Manager component (stage 1 predicates).

ROWS QUALIFIED BY RDS
Number of rows that qualified for the request by DB2's Relational Data Base Support component (stage 2 predicates).

ROWS INSERTED
Number of rows inserted.

ROWS UPDATED
Number of rows updated.

ROWS DELETED
Number of rows deleted.

PAGES SCANNED
Number of data base pages that needed to be scanned to satisfy the request.

REFERENTIAL INTEGRITY PROCESSING:
The data displayed includes:

PAGES SCANNED
Number of pages scanned to enforce referential integrity constraints.

ROWS DELETED/SET NULL
Number of rows deleted or set to a null value to enforce referential integrity.

LOB- PAGES SCANNED
Number of additional pages scanned in a LOB table space.

LOB- PAGES UPDATED
Number of LOB data pages updated by an SQL INSERT or UPDATE.

Database Summary

The summaries shown in Figure 87 can be viewed by scrolling down from the initial STRAC display or by selecting the SCANS expand button. When only the Database Summary section is displayed, it can be sorted by any column (see “STRAC Database Summary Sort” in this section).

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC  INPUT    14:27:46  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,SCAN,SORT=DA    ROW 1 OF 11 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, PKG, SPAS, DDF
         SUMMARIES: SQL, SCANS, IO/LOCK, SORTS
- - - - - - - - - - - DATABASE SUMMARY (DETAIL TRACE ONLY) - - - - - - - - - -

                        ------ INDX -----  ------ DATA -----  ------ WORK+ ----
DATABASE  PAGESET OBID  #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 87. STRAC Database Summary Section

Sort: A qualified Database Summary display can be sorted by any column. To request a sort of the qualified STRAC Database Summary display, type the following command in the STRAC PARM field:

```
,SORT=value
```

where value can be as follows:

DA Database/pageset/OBID - alphabetic ascending sort

This is the default sort.

IS Index scans
IP Index pages scanned
IR Index rows scanned
DS Data scans
DP Data pages scanned
DR Data rows scanned
WS Work scans
WP Work pages scanned
WR Work rows scanned

All numeric sorts are in descending order.

Description: This section is not shown for a summary trace. The heading line is shown with no data if the display is for a detail trace that does not include scans.

DATABASE
Name of the data base.

PAGESET
Name of the page set.

OBID
Five-digit data object identifier. The OBID can be matched with either the SYSIBM.SYSTABLES or SYSIBM.SYSINDEXES table to translate the table or index name.

There is a statistics column with the following subheadings for each type of processing: index, data, or work (DSNDB07). The WORK+ column includes counts from temporary tables and transition tables, in addition to work files.

#SCAN
Number of index, data, or work scans performed.

#PAGE
Number of index, data, or work pages accessed which includes the pages scanned for referential integrity.

#ROWS
Number of rows processed which includes the rows processed for referential integrity.

Database Lock and I/O Summary

The following summaries for a single transaction, shown in Figure 88, can be viewed by scrolling the STRAC display or by selecting the IO/LOCK expand button.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC  INPUT    14:27:53  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,IOLOCK,SORT=DP  ROW 1 OF 11 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, 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
-------- -------- ----- ----- ------  ----- ----- -------- -------- ----- -----
B1C0DB   BB040A12     0     0      0      1     0    21 ms    21 ms     0     0
B2C0DB   BB051012     2     0      0      1     0    18 ms    18 ms     0     0
B3C0DB   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 88. STRAC Database Lock and I/O Summary Section

Sort: A qualified Database Lock and I/O Summary display can be sorted by any column. To request a sort of the qualified STRAC Database Lock and I/O Summary display, type the following command in the STRAC PARM field:

```
,SORT=value
```

where value can be as follows:

DP Database/pageset - alphabetic ascending sort

This is the default sort.

LM Maximum number of locks
LS Lock suspensions
LT Lock timeouts
SR Synchronous I/O - READs
SW Synchronous I/O - WRITEs
SE Synchronous I/O - elapsed time
SA Synchronous I/O - average elapsed time
AR Asynchronous I/O - READ requests
AP Asynchronous I/O - pages read

All numeric sorts are in descending order.

Description: This section is not shown for a summary trace. The heading line is shown with no data if the display is for a detail trace that does not include these events.

These summaries are as follows:

LOCKS
The following statistics are displayed for each data base and page set combination:

Note: Zero values in the DATABASE or PAGESET field indicates the suspension or lock request is an internal latch to access a root index page.

MAX.
Maximum number of locks held.

SUSP.
Number of lock suspensions.

TM/OUT
Number of lock timeouts.

SYNC. I/O
A row of statistics is displayed for each data base and page set combination. This data is available only if data collection of I/O events was specified when the trace request was started. If no I/O events were collected, N/A is displayed in these fields.

READ
Number of synchronous READ I/Os.

WRITE
Number of synchronous WRITE I/Os.

ELAPSED
Total elapsed time for synchronous I/Os.

AVG.ELAP
Average elapsed time per synchronous I/O request.

ASYNC I/O
A row of statistics is displayed for each data base and page set combination. This data is available only if data collection of I/O events was specified when the trace request was started. If no I/O events were collected, N/A is displayed in these fields.

READ
Number of asynchronous READ requests. These can be sequential prefetch, list prefetch, or dynamic sequential detection.

PAGES
Number of pages read asynchronously. Multiple pages can be read with one async READ request.

Sort Summary (Detail Trace Only)

The sort summaries shown in Figure 89 can be viewed by scrolling to the end of the STRAC display or by selecting the SORTS expand button.

Note: This section is not shown for a summary trace. The heading line is shown with no data if the display is for a detail trace that does not include these events.

```
BMC Software --------------  SUMMARY TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> STRAC           INPUT    14:27:16  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> PBCR02,SEQ=5,SORTS                          ROW 1 OF 18 SCROLL=> CSR
EXPAND:  MON(WKLD), DETAIL, HISTORY
         ACCOUNTING: ENV, ELAPSED, SQLCOUNTS, BPOOL, LOCKS, PRLL, 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 89. STRAC Sort Summary Section

This area of the display contains statistics for internal sorts done by DB2 for a single transaction.

NUMBER OF SORTS
: Total number of internal sorts done.

AVG # WORK FILES
: Average number of sort work files used.

The average, maximum, minimum, and total values are calculated for the following:

SORT ELAPSED
: Elapsed time for sort.

SORT RECORDS
: Number of records sorted.

RECORD SIZE
: Size of records sorted.

Chapter 14. Viewing All Events for a Specific Transaction

To view all events for a specific transaction, expand from STRAC to the DTRAC display.

You also can access all the traced events of the current active thread for a specific user (UTRAC) by expanding from the DUSER display.

DTRAC—Detail Trace Entry

The DTRAC Trace Entry Display shows all the traced events in chronological order for a specific transaction entry in the trace buffer. Three levels of display (shown in Figure 90, Figure 91, and Figure 92) can be specified. The levels are 1, 2, or 3, which are typed in the LEVEL= field of the DTRAC display. Each succeeding level shows more detail trace events.

```
BMC Software --------------  DETAIL TRACE ENTRY  --------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:20:10  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TEST3,SEQ=000017,LEVEL=1             ROW     1 OF    22 SCROLL=> CSR
EXPAND:  LINESEL(DETAIL), HISTORY
START: 09:17:11 AUTH: JEK1     PLAN: DSNTIB21 CORR: DB221REQ     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
PREPARE   350     2.069 1,361 ms    26 ms *RC(   0) C=DT         D/X   PS(  10)
OPEN      524     3.489   204 us   202 us *RC(   0) C=DT
FETCH  (5)532     3.489 2,058 ms 6,247 us *RC(   0) C=DT         D/X   PS(   2)
CLOSE     564     5.701    93 ms   809 us *RC(   0) C=DT
PREPARE   350     5.925   656 ms    31 ms *RC(   0) C=DT         D/X   PS(  22)
OPEN      524     6.635   198 us   196 us *RC(   0) C=DT
FETCH     532     6.636 2,239 ms    10 ms *RC(   0) C=DT         D/X   PS(   5)
```

Figure 90. Detail Trace Display for a Transaction (Level 1) - User Requests

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:21:48  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TEST3,SEQ=000017,LEVEL=2             ROW     5 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
-------------- -------- -------- -------- -------------------------------------
PREPARE   350     2.069 1,361 ms    26 ms *RC(   0) C=DT         D/X   PS(  10)
 BIND-TEXT        2.070                   *TYPE=DYNAMIC  TEXT=SELECT * FROM DS+
 EXPLAIN          2.078                   *PLAN=DSN8IC22 COST(4.6)
 EDM-REQ          3.302   123 ms 3,964 us  DB=00000258
OPEN      524     3.489   204 us   202 us *RC(   0) C=DT
FETCH     532     3.489 2,058 ms 6,247 us *RC(   0) C=DT         D/X   PS(   2)
 OPEN-TS          4.235                    DB=DSN8D21A TS=DSN8S21D
 OPEN-TS          5.352                    DB=DSN8D21A TS=XDEPT3
FETCH     532     5.549    35 ms   441 us *RC(   0) C=DT         D/X
FETCH     532     5.615   354 us   353 us *RC(   0) C=DT         D/X
FETCH     532     5.616   335 us   335 us *RC(   0) C=DT         D/X
FETCH     532     5.620   354 us   353 us *RC(   0) C=DT         D/X
FETCH     532     5.621   386 us   387 us *RC(   0) C=DT         D/X
FETCH     532     5.672   386 us   387 us *RC(   0) C=DT         D/X
FETCH     532     5.673   337 us   337 us *RC(   0) C=DT         D/X
```

Figure 91. Detail Trace Display for a Transaction (Level 2) - DB2 Requests

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    08:45:42  INTVL=> 5  LOG=> N  TGT==> DB0G
PARM ==> TEST410,SEQ=1,LEVEL=3                      ROW 59 OF 88 SCROLL=> CSR
EXPAND:  LINESEL(DETAIL)                                        HIST TGT--- DB2G
START: 17:59:40 AUTH: BOLHHH4  PLAN: DSNESPRR CORR: BOLHHH4      CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- ------------------------------------
 LOCK            14.397              0 us *DB=DSNDB06              TYP=PAGESET
 INDX-SCAN       14.484 00:01:15 8,136 us *TS=SYSDBAUT TB=SYSDBAUTH
  CLAIM          14.513                   *DB=DSNDB06  OB=DSNADH01 PART=0
  LOCK           14.572              0 us *DB=DSNDB06  OB=SYSDBAUT TYP=DATAPAGE
  LOCK-SUSP      14.572 00:01:14   352 us *DB=DSNDB06  OB=SYSDBAUT TYP=DATAPAGE
  WAIT-SERVER    88.744                    RMID=MESSAGE GENERATOR (MG)
  WAIT-SERVER    88.746                    RMID=MESSAGE GENERATOR (MG)
  LOCK-TMO-DET   89.114                   *DB=DSNDB06  OB=SYSDBAUT NUMBER=1
  LOCK           89.152              0 us *DB=DSNDB06  OB=SYSDBAUT TYP=DATAPAGE
  WAIT-SERVER    89.374                    RMID=MESSAGE GENERATOR (MG)
  WAIT-SERVER    89.376                    RMID=MESSAGE GENERATOR (MG)
 UNLOCK          89.942              0 us *IRLM TOKEN=00000000
ABORT            90.047   551 ms 1,905 us
 WAIT-SERVER     90.065                    RMID=RECOVERY MANAGER (RM)
  UNLOCK         90.124              0 us *DB=         OB=         TYP=INDEXPAG
  UNLOCK         90.581              0 us *IRLM TOKEN=061B618C
  CLAIM          90.581                   *DB=DSNDB06  OB=DSNADH01 PART=0
  CLAIM          90.581                   *DB=DSNDB01  OB=DSNSPT01 PART=0
  CLAIM          90.582                   *DB=DSNDB01  OB=SPT01    PART=0
  LOCK-SUMMARY   90.582                   *MAXPG(1) ESCL(0) TS(  3)
  COMMIT-LSN     90.582                   *LOCK-AVOID=N PAGESETS=3
  UNLOCK         90.596              0 us *IRLM TOKEN=00000000
SQL-ERROR  19    90.598                   *RC( 692)
SYNC            123.435    36 ms   900 us
 WAIT-SERVER    123.435                    RMID=RECOVERY MANAGER (RM)
  UNLOCK        123.436              0 us *IRLM TOKEN=00000000
COMMIT/DES232   123.471                   *RC(   0)
TERM-THD        123.471              0 us
WAIT-SERVER     123.472                    RMID=RECOVERY MANAGER (RM)
 UNLOCK         123.472              0 us *IRLM TOKEN=00000000
```

Figure 92. Detail Trace Display for a Transaction (Level 3) - DB2 Processing Events

Select Code:

DTRAC

Parameter:

The following parameters can be specified:

trace id

Unique ID identifying the trace. It must match the ID of the request that started the trace. This parameter is positional and a comma delimits the next parameter. If the ID is blank, a comma must precede any other parameter.

LEVEL=1|2|3

The default is 2. Levels 1 or 3 can be typed in this field to change the display at any time. The entries shown at the three levels depend upon the type of detail trace started (see Appendix B, “IFCID Table (ATRAC Data Collection)” on page 409 and “Event Data” on page 259).

Note: There are several types of data collection levels that can be started with an ATRAC request (D, SQL, SCAN, IO, LOCK, DDF, or DDFVTAM), but there are three display levels. This means the correct type of trace must be started to see all of the events. A warning message is displayed on the fourth line of the display if all possible data is not available.

SEQ=
Unique internal sequence number assigned to each trace entry. The number is assigned sequentially and wraps at 32768. If DTRAC is selected from the STRAC display by cursor positioning within the STRAC display, the transaction displayed is identified in the SEQ field. If DTRAC is selected from the Analyzer Display Service list application, the first active entry in the trace buffer is displayed and identified in the SEQ field. Any valid trace buffer entry can be displayed by typing its trace sequence number in the SEQ field. Also see “Scrolling” below.

PGM=
Program name. Only the statements for the specified program are shown. All events subordinate to the selected SQL statements are also displayed.

Description:
Displays all the traced events for one transaction at three succeeding levels by using the LEVEL= parameter field. These display levels provide the following:

LEVEL=1
Displays only those events related to user requests (SQL statements) plus any exception conditions such as authorization failures, timeouts, or deadlocks.

LEVEL=2
Displays level 1 events plus events related to DB2 requests such as CREATE THREAD, EDM requests, or sorts. This display is the default and is the initial display shown when DTRAC is invoked.

LEVEL=3
Displays level 1 and level 2 events plus events related to DB2 processing caused by the user and DB2 requests such as scan, I/O, and lock/latch activity.

Note: If the target for a historical trace is not equal to the current TGT field, it is displayed in the HIST TGT field.

Scrolling:
Each display can be scrolled up (PF7/19) or down (PF8/20). Data for only the current entry in the trace buffer is scrolled. A new transaction in the trace buffer can be displayed by using the left and right PF scroll keys. PF10/22 (left) moves backward through the transaction entries in the buffer. PF11/23 (right) moves forward through the transaction entries. The transaction displayed is identified in the SEQ field of the display.

Expand:
Pop-up displays of additional statistical information can be selected by positioning the cursor anywhere on the line of an entry that has an asterisk (*) in the DETAIL column and pressing ENTER.

An asterisk (*) preceding a field in the DETAIL column means a pop-up display is available for that event.

In addition, you can EXPAND to the following displays when available:

HISTORY
Historical trace data set for this trace (if you are viewing a current trace with logging, this provides access to data no longer in the online buffer).

CURRENT
Active trace data still in the buffers (if you are viewing a trace log data set for a still-active trace or a complete trace that has not yet been purged).

Display Fields

Each display level shows the detail trace data in the fields shown in Figure 93.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:20:10  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TEST3,SEQ=000017,LEVEL=1             ROW     1 OF    22 SCROLL=> CSR
EXPAND:  LINESEL(DETAIL), HISTORY
START: 09:17:11 AUTH: JEK1     PLAN: DSNTIB21 CORR: DB221REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
PREPARE   350     2.069 1,361 ms    26 ms *RC(   0) C=DT         D/X   PS(  10)
```

Figure 93. DTRAC Display Fields

DTRAC displays each traced event in a row. Subordinate events are indented by event name to a maximum of four. The data collected is shown by the following:

Field Name	**Description**
EVENT	Type of event (see "Event Data" on page 259).
AT	Time in seconds relative to the CREATE THREAD time.
ELAPSED	Elapsed time for the event if applicable.
CPU	CPU time for the event if applicable.
DETAIL	Detail data specific to the type of event. An asterisk (*) preceding this field indicates a pop-up display is available that shows additional data.

Note: The ELAPSED and CPU time adjusts to us (microseconds), ms (milliseconds), or hh:mm:ss (hours, minutes, seconds) depending upon the amount of time measured.

Group SQL

Per a user's ATRAC request (GROUPSQL=Y|N), multiple SQL statements can be merged into one displayed line with no intervening traced events by the following:

Field Name	Description
Event	Grouped SQL statements with a numeric indicator (n) as to the number of grouped events.
AT	Time in seconds relative to the first event.
ELAPSED	Total of all elapsed events.
CPU	Total of all CPU time for individual events.
DETAIL	If any value (except for PAGS SCAN total) changes and if that value stops a grouping, that value is shown.

Event Data

The displayed events are a set of IFCIDs collected by the ATRAC service. Appendix B, "IFCID Table (ATRAC Data Collection)" on page 409 lists the IFCIDs that can be requested by ATRAC with each collection category. Each event traced has different types of data associated with it, depending on the requested ATRAC collection category and the DTRAC display level as shown in the following IFCID table. Events with paired IFCIDs can have elapsed and CPU times.

An asterisk (*) in the DETAIL field for a DTRAC event indicates additional data about that event can be shown in a pop-up display. The additional event data is described in the IFCID table below. Each available pop-up display is shown and described in "Event Pop-Up Displays" on page 274.

The following table summarizes events collected for a detail trace. It shows the collection category required to include an event in the trace (TYPE=xxx) and the level at which that event is displayed (DTRAC display level). An asterisk (*) indicates a pop-up display is available for that event.

Table 9. Detail Trace Event Data

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
ABORT	D	1	068/069	Rollback thread.
ASSOC-LOCATR*	SQL	2	272	ASSOCIATE LOCATORS statement (DB2 version 5 and later) (see "Associate Locators Pop-Up (IFCID 272)" on page 276). STPROC= Stored procedure name.
ALLOC-CURSOR*	SQL	2	273	ALLOCATE CURSOR statement (DB2 version 5 and later) (see "Allocate Cursor Pop-Up (IFCID 273)" on page 275). STPROC= Stored procedure name.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
AUTH-FAIL*	D	1	140	Authorization ID not privileged to perform requested operation (see "Authorization Failure Pop-Up (IFCID 140)" on page 277). Type of request: START RLF — STOP RLF DISPLAY RLF — MONITOR 1 MONITOR 2 — CHECK UTILITY SELECT — INSERT DELETE — UPDATE CREATE INDEX — DBADM ALTER — DISPLAY THREAD EXECUTE — BIND/REBIND/FREE CREATEDBA — CREATE STOGROUP DBCTRL — DBMAINT RECOVER INDOUBT — DROP COPY — LOAD EXPLICIT QUALIFIER USE — REORG REPAIR — START DB START/STOP DB2/DB(*) — RUNSTAT STOP DB — START/STOP TRACE SYSADM — SYSOPR USE — BIND ADD RECOVER (UTILITY) — CREATE DBC RECOVER BSDS — CREATE TABLE CREATE TABLESPACE — DISPLAY UTILITY COMMENT ON — LOCK TABLE DISPLAY DB — CREATE SYNONYM ALTER INDEX — DROP SYNONYM DROP INDEX — STOPSPACE UTILITY CREATE VIEW — TERM UTILITY DISPLAY BUFFERPOOL — ALTER BUFFERPOOL SET ARCHIVE — DISPLAY ARCHIVE REFERENCES — TERM UTIL ON DB PACKADM — RENAME TABLE SET ARCHIVE — DISPLAY ARCHIVE START PROCEDURE — STOP PROCEDURE DISPLAY PROCEDURE — CREATE GLOB TEM TBL ALTERIN — START CREATIN — STOP DROPIN — DISPLAY USAGE
AUTH-XLAT*	D	2	169	Authorization ID translation either OUTBOUND or INBOUND (see "Authorization Translation Pop-Up (IFCID 169)" on page 278). OLD= ID before translation. NEW= ID after translation.
BFR-SND BFR-RCV	DDFVTAM	3	166	Buffer send or receive. LU= Logical unit name. SIZE(999) Size of buffer in bytes.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
BIND*	D	1	108/109	Bind/rebind a plan or package (see "Bind Pop-Up (IFCID 108/109)" on page 279). PLAN= Plan name. PACKAGE= Package name. RC(nnn) Return code.
BIND-TEXT*	SQL	2	063	Dynamic/static SQL BIND (see "Bind Text Pop-Up (IFCID 063)" on page 283). TYPE= DYNAMIC or static. Type of language when static: ASM COBOL C FORTRAN PL/1 COBOL II TEXT= First 17 bytes of text of SQL statement. Full text is available in a pop-up display.
CLAIM*	LOCKS	3	211	Claim request (see "Claim Pop-Up (IFCID 211)" on page 284). DB= Database name. OB= Table space or index space name. PART= Partition number.
CLAIM-WAIT*	D	3	215/216	Wait for pending claims to go to zero (see "Claim Wait Pop-Up (IFCID 215/216)" on page 286). DB= Database name. OB= Table space or index space name. PART= Partition number.
CLOSE-TS	D	2	107	Close table space data set. TS= Table space name. DB= Database name.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information

COMMIT-LSN*	D	2	218	Summary of COMMIT log sequence numbers (see "COMMIT-LSN Pop-Up (IFCID 218)" on page 287).
				
				LOCK AVOID= Y for successful lock avoidance technique for unit of work
				N for unsuccessful
				PAGESETS= Number of pagesets involved in unit of work
				TYPL= Type of lock:
				`PAGE`
				`ROW`
				`IXEOF (index end-of-file)`

CONV-QUEUED*	D	2	167	Conversation request queued (see "Conversation Queued Pop-Up (IFCID 167)" on page 288 for pop-up display). LU= Logical unit name. MODE= VTAM mode name from communications database.

CONV-REQ*	DDF	2	159	Conversation request (see "Conversation Request Pop-Up (IFCID 159)" on page 289 for pop-up display).
				
				TYPE= Type of request:
				`CREATE`
				`WAIT`
				LOC= Sixteen-character location name.

CREATE-THD	D	2	072/073	CREATE THREAD.
DEADLOCK*	D	1	172	Deadlock details (see "Deadlock Pop-Up (IFCID 172)" on page 290 for pop-up display). RESOURCES INVOLVED= Number of locked resources in the deadlock.
DIST-SQL*	D	2	168	Distributed SQL statement (see "Distributed SQL Pop-Up (IFCID 168)" on page 293 for pop-up display).

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
DRAIN*	LOCKS	3	212	Drain request (see "Drain Pop-Up (IFCID 212)" on page 294). DB= Database name. OB= Table space or index space name. PART= Partition number.
DRAIN-WAIT*	D	3	213/214	Wait for drain request (see "Drain Wait Pop-Up (IFCID 213/214)" on page 296). DB= Database name. OB= Table space or index space name. PART= Partition number.
EDM-FULL	D	1	031	EDM pool full condition. Type of request that received condition. CT= Plan name for cursor table. XT= Plan name for cursor table extension. SKCT-HDR Skeleton cursor table header. SKCT-DIR Skeleton cursor table directory. DB= Database name; zeros in this field indicate an internal latch. PT= Package name. SKPT Skeleton package table entry. SKPT-DIR Skeleton package table directory.
EDM-REQ	D	2	029/030	EDM I/O Request. CT= or XT= The plan name for the cursor table or extended cursor table respectively. SKCT-DIR Skeleton cursor table directory. SKCT-HDR Skeleton cursor table header. RDS-SECTION Cursor table sections. DB= Database name. Zeros in this field indicate an internal latch. PT= Collection ID and package ID.
End SQL*: COMMIT/DESC or SQL-ERROR	SQL	1	053	End DESCRIBE, COMMIT, ROLLBACK, or SQL statement error (see "SQL Statement Pop-Up" on page 356).

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
EXPLAIN*	SQL	2	022	Explain the data access path for an SQL statement (see "EXPLAIN Pop-Up (IFCID 022)" on page 298). PLAN/PKG= Plan name or package ID. COST(n.n) The relative cost assigned by the DB2 optimizer for the execution of the SQL statement; for large numbers, a unit of measure (K, M, or B) is appended.
FREE*	D	1	110/111	Free a plan or package (see "Free Pop-Up (IFCID 110/111)" on page 314). PACKAGE= Package name. PLAN= Plan name. RC(nnn) Return code.
FRST-INSRT (n)*	SCAN	3	016/018	Internal INSERT of rows (see "Scan Pop-Up (IFCID 017/018)" on page 348) where (n) identifies the number of grouped scan entries, if any. IX= Index name. TS= Table space name. DB= Database name.
HASH-SCAN	SCAN	3	013/014	Hash scan. TS= Table space name. DB= Database name.
HJOIN-FAIL*	D	2	190	Hybrid join failure (see "Hybrid Join Failure Pop-Up (IFCID 190)" on page 315). This is a single-event record with no elapsed or CPU time. #RIDS= Number of record ID entries (RIDs) in the RID list. type-of-failure The type of failure described in a short phrase.
INDX-SCAN (n)*	SCAN	3	015/018	Matching or non-matching index scan (see "Scan Pop-Up (IFCID 017/018)" on page 348) where (n) identifies the number of grouped scan entries, if any. IX= Index name. TS= Table space name. DB= Database name.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
INV-SELECT	D	1	224	Invalid SELECT procedure. COL=nnn Total columns bypassed, followed by plan name, package ID, or collection ID.
LATCH WAIT	D	3	052/051 056/057	SHARED. EXCLUSIVE.
LOCK* UNLOCK* LOCK CHG* LOCK QRY*	LOCKS	3	021	Lock request (see "Lock Request Pop-Up (IFCID 021)" on page 316). OB= Table space or index space name. DB= Database name. TYP= Type of lock (same as LOCK-SUSP). **Note:** If there is no OB or DB information, the IRLM request token is displayed. PLAN= Plan name. TYP= Type of lock (same as LOCK-SUSP).
LOCK-SUMMARY*	D	2	020	Lock summary at COMMIT/ABORT (see "Lock Summary Pop-Up (IFCID 020)" on page 318). MAXPG(nnn) Maximum page locks held concurrently. ESCL(nnn) Total lock escalations. TS(nn) Number of table spaces in which locks were held.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information

LOCK-SUSP*	D	3	044/045	Lock suspension (see "Lock Suspension Pop-Up (IFCID 044/045)" on page 320).
				
				TYP= Abbreviated lock type (see "Lock Type Table" on page 405 for a complete description).
				
				Type of lock:
				
				`ALTER BP` `OPEN`
				`BINDLOCK` `PAGE PLK`
				`CATM CAT` `PAGESET`
				`CATM DIR` `PART-DS`
				`CATM MIG` `P/P CAST`
				`CDB PLOK` `P/P PLOK`
				`COLLECTN` `REPR DBD`
				`CS-DRAIN` `RLF PLOK`
				`DATABASE` `ROW`
				`DATAPAGE` `RR-DRAIN`
				`DBALLOC` `SCA ACC`
				`DBD PLOK` `SERVICE`
				`GBP CAST` `SKCT`
				`GP EX UP` `SKPT`
				`GROUP BP` `SYSLGRNG`
				`HASH-ANC` `TABLE`
				`INDEXPAG` `UTIL EXC`
				`IX EOF` `UTILSER`
				`IXTREEPL` `UTILUID`
				`LPL/GREC` `WR-DRAIN`
				`MASSDEL`
				
				OB= Table space or index space name.
				DB= Database name.
				
				Note: If there is no OB or DB information, the IRLM request token is displayed.

LOCK-TMO*	D	1	054	Lock timeout/deadlock (see "DTRAC Lock Timeout/Deadlock Pop-Up Display" on page 322). TYPE= ACE for another user or URE for unit of recovery element. **Note:** This event does not exist for DB2 version 4 and later (see TIMEOUT and DEADLOCK events).
LSN-DETAIL	LOCKS	3	223	Detail of successful COMMIT-LSNs (lock avoidance). OB= Table space or index space name. DB= Database name.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
OPEN-TS	D	2	107	Open table space data set. TS= Table space name. DB= Database name.
PAGE P-LOCK*	LOCKS	2	259	Page physical lock (P-lock) request (see "DTRAC Page P-Lock Pop-Up Display" on page 323). DB= Database name. PS= Page set name. IRLM function LOCK, UNLOCK, CHANGE, or EXITCH (change from P-lock exit). If the field is blank, it does not display.
PGLAT-WAIT	D	1	226/227	Page latch wait. OB= Table space or index space name. DB= Database name. POOL= Buffer pool ID.
PH1-COMMIT	D	2	084/085	Phase 1 commit.
PH2-COMMIT	D	2	070/071	Phase 2 commit.
PKG-ALLOC*	D	2	177	Package allocated (see "Package Allocation Pop-Up (IFCID 177)" on page 325). ISO= Type of isolation, either CS (cursor stability), RR (repeatable read), or UR (uncommitted read). ACQ= Acquire locks at ALLOC (allocation) or USE (first use). REL= Release level of the package, determining when resource are released; possible values are DEALLOC (deallocation) or COMMIT.
PLAN-ALLOC*	D	2	112	Plan allocated (see "Plan Allocation Pop-Up (IFCID 112)" on page 334). ISO= Type of isolation, either CS (cursor stability) or RR (repeatable read). ACQ= Acquire locks at ALLOC (allocation) or USE (first use). REL= Release locks at DEALLOC (deallocation) or COMMIT.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
P-LOCK*	LOCKS	2	251	Physical lock (P-lock) operation (see "P-Lock Pop-Up (IFCID 251)" on page 335). DB= Database name. PS= Page set name. IRLM function LOCK, UNLOCK, CHANGE, or EXITCH (change from P-lock exit). If the field is blank, it does not display.
PRLL-DEGREE*	SCAN	3	221	Degree of parallel I/O processing for group (see "Read I/O Pop-Up (IFCID 006/007)" on page 339). PLANNED= Degree planned at run time. ACTUAL= Actual parallel I/O degree. GROUP#= Parallel group number.
PRLL-ELAP*	SCAN	3	222	Elapsed time of parallel group (see "Parallel Elapsed Pop-Up (IFCID 222)" on page 330). ELAP= Elapsed time of parallel group pipe. ROWS= Number of rows processed.
PRLL-GROUP*	SCAN	3	231	Individual task information for a parallel group (DB2 5.1 and later) (see "Parallel Group Pop-Up (IFCID 231)" on page 332). ELAP= Elapsed time of parallel group. GROUP= Parallel group number.
READ-I/O*	IO	3	006/007	READ I/O (see "Parallel Group Pop-Up (IFCID 231)" on page 332). DB= Database name. TS= Table space name. BPnnnn Buffer pool in use. One of four possible descriptions is reported: SYNC-RD Synchronous read of one page. SEQ-PF Asynchronous read for sequential prefetch. LIST-PF Asynchronous read for list prefetch. DYN-SEQ Asynchronous read for dynamic sequential detection prefetch.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information

REQ-AGNT*	DDF	2	183	Requesting agent data (see "Requesting Agent Data Pop-Up (IFCID 183)" on page 340).
				RC Return code from event.
				function The function type; possible values are:
				`REBIND`
				`FREE`
				`BEGIN BIND`
				`BIND STMT`
				`END BIND`
				`SQL STMT`
				`CONNECT EXTRACT`
				`CONNECT RESET`
				`IMPLICIT CONNECT`
				`EXPLICIT CONNECT`
				`BIND CONNECT`
				`DISCONNECT`
				`CONNECT TYPE 2`
				`SET CONNECTION`
				`RELEASE`

REQ-DRDS*	DDF	2	157	DRDS request at requesting site (see "Requesting Site—DRDS Pop-Up (IFCID 157)" on page 342).
				TYPE= Type of request:
				`AUXCALL`
				`SETUPCAL`
				`DSCRBCAL`
				`OPENCALL`
				LOC= Sixteen-character location name.

REQ-DTM*	DDF	2	162	Distributed Transaction Manager request at requesting site (see "Requesting Site—DTM (IFCID 162)" on page 343).
				TYPE= Type of request:
				`MIGRATE TRANSACTION`
				`COMMIT MESSAGE SENT`
				`DEALLOCATION INITIATED`
				`READ RESPONSE RECV'D`
				`BACKOUT REQUEST SENT`
				`COMMIT RESPONSE RECV'D`
				`BACKOUT RESPONSE RECV'D`
				`PREPARE REQUEST SENT`

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information

REQ-VTAM*	DDF	3	160	VTAM request at requesting site (see "Requesting Site—VTAM Pop-Up (IFCID 160)" on page 344).
				TYPE= Type of request:
				`SEND REQUEST MESSAGE`
				`ALLOCATE CONVERSATION`
				`CREATE CONVERSATION`
				`DEALLOCATE CONVERSATION`
				`RESET CONVERSATION`
				`RECEIVE RESPONSE MESSAGE`
				`TERMINATE CONVERSATION`
				`WAIT FOR RESPONSE MESSAGE`

RIDLIST*	D	2	125	Multiple index access path processing of RID lists (see "RID List Pop-Up (IFCID 125)" on page 346).
				One of three possible descriptions is reported:
				`RIDS IN FINAL LIST (nnn)`
				`NOT USED - NO RID STORAGE`
				`NOT USED - RIDS > MAXIMUM LIMIT`

SEQ-SCAN (n)*	SCAN	3	017/018	Sequential table space or work file scan (see "Scan Pop-Up (IFCID 017/018)" on page 348) where (n) identifies the number of grouped scan entries, if any. TS= Table space name. DB= Database name.

SERV-DRDS*	DDF	2	158	RDS request at serving site (see "Serving Site—DRDS Pop-Up (IFCID 158)" on page 350).
				TYPE= Type of request:
				`AUXCALL`
				`SETUPCAL`
				`DSCRBCAL`
				`OPENCALL`

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information

SERV-DTM*	DDF	2	163	Distributed Transaction Manager request at serving site (see "Serving Site—DTM Pop-Up (IFCID 163)" on page 351).
				TYPE= Type of request:
				`MIGRATE TRANSACTION RECEIVED`
				`COMMIT MESSAGE RECEIVED`
				`DEALLOCATION INITIATED`
				`RESPONSE SENT`
				`(READ LOCATION)`
				`ACKNOWLEDGE COMMIT SENT`
				`(UP LOCATION)`
				`NEW USER BROADCAST RECEIVED`
				`DBAA CREATION QUEUED`
				`BACKOUT REQUEST RECV'D`
				`NEW USER BROADCAST`
				`DB ACCESS AGENT CREATE Q'D`
				`COMMIT RESPONSE SENT`
				`BACKOUT RESPONSE SENT`
				`PREPARE REQUEST RECV'D`
				`LAST AGENT REQUEST RECV'D`

SERV-VTAM*	DDF	3	161	VTAM request at serving site (see "Serving Site—VTAM Pop-Up (IFCID 161)" on page 352).
				TYPE= Type of request:
				`RECEIVE CONVERSATION`
				`DEALLOCATE CONVERSATION`
				`RECEIVE REQUEST MESSAGE`
				`SEND RESPONSE MESSAGE`

SET DEGREE	D	1	237	Set current degree. DEGREE= OLD|NEW, where OLD is previous degree and NEW is attempted degree. Only valid values are 1 or ANY. SUCCESS= Y for yes. N for no.
SET SQLID	SQL	1	055	SET SQLID. STATUS= S (successful), F (failed), or X (successful with SYSADM AUTH). TO= New SQL ID. FROM= Old SQL ID.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
SORT*	D	2	095/096	DB2 internal sort (see "DTRAC Sort Pop-Up Display" on page 354). REC(nnn) Number of records. RECSZ(nn) Record size. WKFILE(n) Number of work files.
SQL* FETCH SELECT INSERT/UPDATE/ DELETE DDL PREPARE OPEN CURSOR CLOSE CURSOR	SQL	1	 059/058 060/058 061/058 062/058 064/058 065/058 066/058	Contains the SQL statement type followed by the SQL precompiler statement number (see "SQL Statement Pop-Up" on page 356). For grouped SQL statements, the number of grouped events is shown in parentheses in front of the statement number. The statement number has a unique character appended to it to identify the program. The first program is assigned a blank. Additional programs for the same plan are assigned the characters A - Z; for example, FETCH 120A. The program name can be seen in the pop-up display (a series of periods (.) indicates an unidentifiable program name). RC(snnn) The SQL return code for the statement. C= The cursor name, if applicable. D Data scan was done. X Index scan was done. W Work scan was done. PS(nnnn) Pages scanned; includes pages scanned for referential integrity. object type For DDL, the object type (TABLE) followed by the table name.
STORPROC*	SQL	2	233	Stored procedure CALL statement (see "Stored Procedures Entry Pop-Up (IFCID 233)" on page 359 and "Stored Procedures Exit Pop-Up (IFCID 233)" on page 360). MODE ENTRY (upon entry to the stored procedure) or EXIT (upon exit back to the caller of the stored procedure). PROC Stored procedure name.
SYNC	D	2	088/089	Synchronization event as part of explicit or implicit commit processing.
SYNC-WRITE*	IO	3	008/009	Synchronous write (see "Synchronous Write Pop-Up (IFCID 008/009)" on page 361). DB= Database name. TS= Table space name. BPnnnn Buffer pool in use.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information
TBL-CONSTR*	SQL	2	305	Table check constraint activity (see "Table Check Constraint Pop-Up (IFCID 305)" on page 362). DB= Database name. OB= Table space or index space name.
TEMPTABLE*	SQL	2	311	Temporary table information (DB2 version 5 and later) (see "Temporary Table Pop-Up (IFCID 311)" on page 364). TABLE= Temporary table name.
TERM-THD	D	2	074	Terminate thread.
TIMEOUT*	D	1	196	Lock timeout details (see "Timeout Pop-Up (IFCID 196)" on page 366). DB= Database name. OB= Table space or index space name. NUMBER= Number of lock holders/waiters.
TRIGGER*	SQL	2	325	Trigger activation (DB2 version 6 and later) (see "Trigger Entry Pop-Up (IFCID 325)" on page 369 and "Trigger Exit Pop-Up (IFCID 325)" on page 371). MODE= ENTRY (upon the start of trigger activation) or EXIT (upon the end of trigger activation). TRIG= Trigger name.
UDF*	SQL	2	324	User-defined function resolution information (DB2 version 6 and later) (see "User-Defined Function Pop-Up (IFCID 324)" on page 373). APPL= Application name. PLAN= Plan name.
Utility name*	D	1	023/024 024/024 024/025	Utility name, such as RUNSTATS or RECOVER (see "Utility Pop-Up" on page 375). ID= Sixteen-character utility ID. PH= Phase name.

Table 9. Detail Trace Event Data (Continued)

Event	ATRAC Collection Category	DTRAC Display Level	IFCID	Description and Detail Information

VTAM-EXIT*	DDFVTAM	3	164	VTAM exit entered (see "VTAM Exit Pop-Up (IFCID 164)" on page 377).
				TYPE= Type of request:
				`ATTN|RPL with SUBTYPE=`
				`CNOS`
				`LOSS`
				`FMH5`
				`TPEND with SUBTYPE=`
				`HALT NET`
				`HALT QUICK`
				`HALT CANCEL`

VTAM-MACRO*	DDFVTAM	3	165	VTAM macro issued (see "VTAM Macro Pop-Up (IFCID 165)" on page 378).
				RC(xx) Return code from macro; type of macro:
				`OPEN`
				`CLOSE`
				`SETLOGON`
				`APPCCMD.....`

WAIT-I/O*	IO	3	127/128	Page wait for I/O in progress (see "Wait I/O Pop-Up (IFCID 127/128)" on page 379). DB= Database name. TS= Table space name. BPnnnn Buffer pool in use. SYNC-RD Synchronous read I/O.
WAIT-SERVER	D	3	170/171	Wait for DB2 server task. Records the beginning of a suspend for a synchronous (either related or unrelated) execution unit. RMID= Subcomponent name that corresponds to the Resource Manager the server task is calling. This field may be blank when displaying an old trace data set.

Event Pop-Up Displays

The pop-up displays shown in this section are invoked by placing the cursor on any event row of the DTRAC display that contains an asterisk (*) in the DETAIL field and pressing ENTER. Pressing the END key (PF3/15) returns the previous DTRAC display level. Once a pop-up display is selected, the previous or next event's pop-up display can be viewed by using PF10/22 or PF11/23 respectively.

Allocate Cursor Pop-Up (IFCID 273)

Additional data for the ALLOC-CURSOR event is displayed in the Allocate Cursor pop-up display, as shown in Figure 94.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    08:29:10  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> NEWSQL3,SEQ=1,LEVEL=2                       ROW 1 OF 11 SCROLL=> CSR

START: 07:15:35 AUTH: BOLSMR3  PLAN: DSN8ED1  CORR: BOLSMRE1     CONN: BATCH
===============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
ALLOC-CURSOR      0.170                   *STPROC=DSN8ED2
===============================================================================

                    LOCATION NAME: DB2H
           PACKAGE COLLECTION ID: DSN8ED1
                     PROGRAM NAME: DSN8ED1
    PROGRAM PRECOMPILE TIMESTAMP: 15E4540B0FCDD898
                STATEMENT NUMBER:   757
            ALLOCATE CURSOR NAME: DSN8ED2_RS_CSR
 REAL CURSOR NAME IN STORED PROC: DSN8ED2_RS_CSR
    LOCATION OF STORED PROCEDURE: DB2H
   QUALIFIER OF STORED PROCEDURE: SYSPROC
                   LOCATOR VALUE: 00000001
```

Figure 94. DTRAC Allocate Cursor Pop-Up Display

This event records information about execution of an ALLOCATE CURSOR statement.

The pop-up display shows the following:

LOCATION NAME
: Name of the location where the stored procedure executes.

PACKAGE COLLECTION ID
: Package collection ID. If DB2 does not use a package to execute this statement, this field is blank.

PROGRAM NAME
: Name of the program.

PROGRAM PRECOMPILE TIMESTAMP
: Precompile timestamp of the stored procedure program.

STATEMENT NUMBER
: Statement number of the ALLOCATE CURSOR statement.

ALLOCATE CURSOR NAME
: Name of the cursor in ALLOCATE CURSOR statement.

REAL CURSOR NAME IN STORED PROC
: Name of the cursor in the stored procedure.

LOCATION OF STORED PROCEDURE
: Name of location where stored procedure resides.

QUALIFIER OF STORED PROCEDURE
: Qualifier of the stored procedure.

LOCATOR VALUE
: Value of the locator associated with the result set for which this cursor is defined.

Associate Locators Pop-Up (IFCID 272)

Additional data for the ASSOC-LOCATR event is displayed in the Associate Locators pop-up display, as shown in Figure 95.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    08:28:51  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> NEWSQL3,SEQ=1,LEVEL=2                        ROW 1 OF 8 SCROLL=> CSR

START: 07:15:35 AUTH: BOLSMR3  PLAN: DSN8ED1  CORR: BOLSMRE1     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
ASSOC-LOCATR      0.170                   *STPROC=DSN8ED2
================================================================================

                   LOCATION NAME: DB2H
                    PROGRAM NAME: DSN8ED1
    PROGRAM PRECOMPILE TIMESTAMP: 15E4540B0FCDD898
                STATEMENT NUMBER:   742
    LOCATION OF STORED PROCEDURE: DB2H
   QUALIFIER OF STORED PROCEDURE: SYSPROC
             NUMBER OF LOCATORS: 00001
```

Figure 95. DTRAC Associate Locators Pop-Up Display

This event records information about execution of an ASSOCIATE LOCATORS statement.

The pop-up display shows the following:

LOCATION NAME
: Name of the location where the stored procedure executes.

PROGRAM NAME
: Name of the program.

PROGRAM PRECOMPILE TIMESTAMP
: Precompile timestamp of the stored procedure program.

STATEMENT NUMBER
: Statement number of the ASSOCIATE LOCATORS statement.

LOCATION OF STORED PROCEDURE
: Name of location where stored procedure resides.

QUALIFIER OF STORED PROCEDURE
: Qualifier of the stored procedure.

NUMBER OF LOCATORS
: Number of locators referenced in the ASSOCIATE LOCATORS statement.

Authorization Failure Pop-Up (IFCID 140)

Additional data for the AUTH-FAIL event is displayed in the Authorization Failure pop-up display, as shown in Figure 96.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    16:52:59  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> D1,SEQ=000001,LEVEL=1                ROW     1 OF     6 SCROLL=> CSR

START: 16:51:54 AUTH: CIR10    PLAN: DSNESPRR CORR: CIR10        CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 AUTH-FAIL        1.362                   *SELECT             SYSVOLUMES
==============================================================================
        OBJECT TYPE: TABLE/VIEW
        AUTH ID:     CIR10
        SOURCE OBJ:  SYSIBM  .SYSVOLUMES
        TARGET OBJ:          .
SELECT *
 FROM SYSIBM.SYSVOLUMES
```

Figure 96. DTRAC Authorization Failure Pop-Up Display

This pop-up display shows an AUTHID that is not privileged to perform a requested operation by the following:

OBJECT TYPE:
The type of object checked for access authorization for this authorization ID.

AUTH ID:
The privilege-checked authorization ID.

SOURCE OBJ:
The object creator and the source object of the requested operation.

TARGET OBJ:
The target creator and the target object of the requested operation.

Note: See the AUTH-FAIL entry in Table 9 on page 259 for a list of the possible types of requests shown on the detail line.

Authorization Translation Pop-Up (IFCID 169)

Additional data for the AUTH-XLAT event is displayed in the Authorization Translation pop-up display, as shown in Figure 97.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:18:08  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=2                                ROW 1 OF 5 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 AUTH-XLAT        3.816                   *OUTBOUND OLD=CIR4     NEW=CIR4
================================================================================

DISTRIBUTED AUTHORIZATION TRANSLATION

          SERVING LOCATION NAME: DB1D
          SERVING LU NAME:       LUDB1D
```

Figure 97. DTRAC Authorization Translation Pop-Up Display

This event occurs both before and after alias translation.

The pop-up display shows the following:

SERVING LOCATION NAME
: Name of location serving the request.

SERVING LU NAME
: VTAM logical unit name of location serving the request.

Bind Pop-Up (IFCID 108/109)

Additional data for the BIND event is displayed in the Bind pop-up display, as shown in Figure 98.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    12:01:40  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> SMRBIND1,SEQ=3,LEVEL=2                      ROW 1 OF 22 SCROLL=> CSR

START: 11:51:04 AUTH: BOLSMR4  PLAN: DSNBIND  CORR: BOLSMR2T     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
BIND              0.360    88 ms    27 ms *RC=(   0)  PKG=SMRPKG31
================================================================================
        ISOLATION LEVEL: CS
        ACQUIRE:         FIRST USE
        RELEASE:         DEFAULT
        VALIDATE:        RUNTIME

        TYPE:            BIND
        ACTION:          ADD
        REBIND PLAN(*):  NO
        EXPLAIN:         NO
        DEGREE:          1
        SQLRULES:
        DYNAMIC RULES:   NONE
        DISCONNECT:
        OWNER:           BOLSMR4
        OBJECT TYPE:     PACKAGE
        QUALIFIER:       BOLSMR4
        PKG LOCATION:    DB2G
        COLLECTION ID:   SMRPKG31
        PACKAGE-ID:      SMRTIAD
        TOKEN:           X'156EC4401A678528'
        VERSION:

        DEFER RMTE SQL:  NO
        REOPTIMIZE SQL:  NO
        KEEP DYN STMTS:  YES
        DB PROTOCOL:     DRDA
        OPTIMIZE HINT:
```

Figure 98. DTRAC Bind Pop-Up Display

This pop-up display shows the DB2 bind parameters specified for the BIND command (see the IBM *DB2 Command and Utility Reference* manual for a complete description of these parameters). If the event is BIND PLAN, the display shows the following fields:

ISOLATION LEVEL
ACQUIRE
RELEASE
VALIDATE
TYPE
ACTION
REBIND
EXPLAIN
DEGREE
SQLRULES
DYNAMIC RULES
DISCONNECT

DEFER RMTE SQL
REOPTIMIZE SQL
KEEP DYN STMTS

If the event is BIND PACKAGE, all of the following events are shown:

ISOLATION LEVEL
The isolation level for plans and packages:

UR	UNCOMMITTED READ
RR	REPEATABLE READ
CS	CURSOR STABILITY
RS	READ STABILITY
DF	Default. This value is valid only for packages. The isolation level of the package defaults at run time to the isolation level of the plan it runs under.

ACQUIRE
When the resources are acquired:

FIRST USE	When they are first accessed (default).
ALLOCATE	When the plan is allocated.

RELEASE
When the resources are released:

COMMIT	At each COMMIT (default).
DEALLOCATE	When the application terminates.
F	Defaults to the PLAN parameter (for package binds).

VALIDATE
When validity checking completes:

RUN	Defers some checking until the plan is used (default).
BIND	Checking completed during bind processing.

TYPE
Specifies BIND, REBIND, or automatic BIND.

ACTION
Specifies whether the application plan can be replaced by a new one with the same name (default) or that a new plan is to be added:

REPLACE
ADD

REBIND
Specifies if this plan is a rebind.

EXPLAIN
Specifies whether EXPLAIN information is provided:

NO (default)
YES

DEGREE
Parallel degree BIND option. It can be either 1 (no parallelism) or ANY (parallelism enabled).

SQLRULES
SQLRULES BIND option. It can be either DB2 or STD.

DYNAMIC RULES
DYNAMICRULES BIND option for dynamic SQL statements. It can be one of the following:

BIND Bind-time rules apply for dynamic SQL statements at run time.
RUN Run-time rules apply for dynamic SQL statements at run time.
NONE DYNAMICRULES was not specified (for packages only). The package has the DYNAMICRULES option value of the plan to which it is appended at run time.

DISCONNECT
DISCONNECT BIND option. It can be one of the following:

EXPLICIT
AUTOMATIC
CONDITIONAL

OWNER
Specified owner of the plan or package.

OBJECT TYPE
Type of object bound, either plan or package.

QUALIFIER
Qualifier for unqualified objects.

PKG LOCATION
Package location. This is blank if the local location is unnamed.

COLLECTION-ID
Collection ID of the package.

PACKAGE-ID
Package ID of the package.

TOKEN
Timestamp for the BIND.

VERSION
Version identification name.

DEFER RMTE SQL
YES|NO. YES indicates preparation of dynamic SQL statements is deferred. NO indicates they are prepared immediately.

REOPTIMIZE SQL
YES|NO. NO indicates access paths are chosen only at bind or rebind time. YES indicates they are determined again at execution time.

Access paths can be determined at execution time for statements with the following types of variable input:

- Input host variables
- Parameter markers
- Values in special registers

KEEP DYN STMTS
YES|NO. Indicates whether DB2 keeps (YES) or discards (NO) prepared SQL statements at commit points.

DB PROTOCOL
Database protocol for three-part names, which can be DRDA or PRIVATE.

OPTIMIZE HINT
Optimization hints value, or blanks if no optimization hints are defined.

Bind Text Pop-Up (IFCID 063)

Additional data for the BIND-TEXT event is displayed in the Bind Text pop-up display, as shown in Figure 99.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    12:31:17  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TRACE1,SEQ=000002,LEVEL=3            ROW     1 OF     1 SCROLL=> CSR

START: 12:26:16 AUTH: BABUSER  PLAN: DSNTIA21 CORR: DB221REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
BIND-TEXT         6.380                   *TYPE=DYNAMIC  TEXT=INSERT INTO T32 +
==============================================================================
INSERT INTO T32 VALUES ('A','B','C','D')
```

Figure 99. DTRAC Bind Text Pop-Up Display

A scrollable field is displayed to show the text of the statement being bound. The area is scrollable to accommodate large SQL statements. Multiple embedded blanks are compressed to one blank.

Note: DB2 limits the length of the text that is put into IFCID 63, so this text may be truncated.

Claim Pop-Up (IFCID 211)

Additional data for the CLAIM event is displayed in the Claim pop-up display, as shown in Figure 100.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC            INPUT   11:10:22  INTVL=> 5  LOG=> N  TGT==> DB2F
PARM ==> DB231A,SEQ=3,LEVEL=3                          ROW 1 OF 5 SCROLL=> CSR

START: 08:40:31 AUTH: CJN1     PLAN: CJNTIA31 CORR: CJN1DBX      CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
  CLAIM           0.591                   *DB=DSNDB06  OB=DSNAPH01 PART=0
==============================================================================
             REQUEST TYPE:   ACQUIRE
             CLASS:          CURSOR STABILITY READ
             DURATION:       COMMIT DURATION
             RETURN CODE:       0
             FAIL REASON:    SUCCESSFUL
```

Figure 100. DTRAC Claim Pop-Up Display

This pop-up display shows the following:

REQUEST TYPE
: Request type, which can be one of the following:

  ACQUIRE THE CLAIM
  CHANGE THE CLAIM DURATION
  RELEASE THE CLAIM

CLASS
: CLAIM class, which can be one of the following:

  CURSOR STABILITY READ
  REPEATABLE READ
  WRITE

DURATION
: Duration of the CLAIM, which can be one of the following:

  HELD UNTIL A COMMIT
  HELD ACROSS A COMMIT
  ALLOCATION
  CLAIM IS RELEASED

RETURN CODE
: Result of the CLAIM request.

FAIL REASON

Reason for the unsuccessful CLAIM (if not SUCCESSFUL), which can be one of the following:

- Resource is started for read-only access.
- Resource is stopped.
- Resource is in use by a utility.
- Resource is in use by a utility that allows only read-only access.
- Resource is started for utility-only access.
- Deadlock.
- Timeout.
- IRLM is out of storage.
- Error in IRLM.
- Resource has an image copy pending.
- Resource has a recovery pending.
- Resource has a check pending.

Claim Wait Pop-Up (IFCID 215/216)

Additional data for the CLAIM-WAIT event is displayed in the Claim Wait pop-up display, as shown in Figure 101.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    16:52:59  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> D1,SEQ=000001,LEVEL=3                ROW     1 OF     6 SCROLL=> CSR

START: 16:51:54 AUTH: CIR10    PLAN: DSNESPRR CORR: CIR10        CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 CLAIM-WAIT       1.362       3s          *OB=DSNAPH01 DB=DSNDB06  PART=0
==============================================================================
        CLASS:          CURSOR STABILITY READ
        RESUME REASON:  NORMAL
        CLAIMS PENDING: 2
```

Figure 101. DTRAC Claim Wait Pop-Up Display

This pop-up display shows the following:

CLASS
: Claim class, which can be one of the following:

  CURSOR STABILITY READ
  REPEATABLE READ
  WRITE

RESUME REASON
: Reason for the resume, which can be one of the following:

  TIMEOUT
  NORMAL

CLAIMS PENDING
: Number of claims pending on this resource.

COMMIT-LSN Pop-Up (IFCID 218)

Additional data for the COMMIT-LSN event is displayed in the COMMIT-LSN pop-up display, as shown in Figure 102.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC            INPUT   10:05:04  INTVL=> 5  LOG=> N  TGT==> DB2F
PARM ==> TRAUTH,SEQ=1,LEVEL=2                         ROW 1 OF 5 SCROLL=> CSR

START: 17:06:41 AUTH: T127     PLAN: DSN8CC0  CORR: GT00D8CS     CONN: CICS332B
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
  COMMIT-LSN     15.254                   *LOCK-AVOID=Y PAGESETS=3 TYPL=ROW
==============================================================================
       DATABASE    TABLE SPACE   LOCK AVOIDANCE
       ---------   -----------   --------------
       DSN8D31A    DSN8S31E      YES
       DSN8D31A    DSN8S31D      YES
       DSN8D31P    DSN8S31C      NO
```

Figure 102. DTRAC COMMIT-LSN Pop-Up Display

This pop-up display shows the following:

DATABASE
Database ID of the database to which the table space belongs.

TABLE SPACE
Page set ID (the OBID) in table space column.

LOCK AVOIDANCE
Whether lock avoidance techniques were used for this page during this unit of work:

YES (They were used)
NO (They were not used)

Conversation Queued Pop-Up (IFCID 167)

Additional data for the CONV-QUEUED event is displayed in the Conversation Queued pop-up display, as shown in Figure 103.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:55:24  INTVL=> 3  LOG=> N  TGT==> DB1D
PARM ==> ,SEQ=6,LEVEL=1                              ROW 1 OF 13 SCROLL=> CSR

START: 10:52:41 AUTH: CIR2     PLAN: DSNESPCS CORR: CIR2         CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
CONV-QUEUED       0.991                   *LU=LUDB1D   MODE=IBMDB2LM
==============================================================================

CONVERSATION ALLOCATION REQUEST QUEUED

        CONVERSATIONS ALLOCATED: 20
        CONVERSATIONS QUEUED:    3
        CONVERSATIONS LIMIT:     20
        CONVERSATION ID:         X'031440B8'
```

Figure 103. DTRAC Conversation Queued Pop-Up Display

This pop-up display shows the following:

CONVERSATIONS ALLOCATED
: Number of conversations allocated when this conversation was queued.

CONVERSATIONS QUEUED
: Number of conversations queued at the time.

CONVERSATIONS LIMIT
: Maximum number of conversations allowed.

CONVERSATION ID
: Four-byte logical connection identifier in hexadecimal.

Conversation Request Pop-Up (IFCID 159)

Additional data for the CONV-REQ event is displayed in the Conversation Request pop-up display, as shown in Figure 104.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:18:19  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=2                                ROW 1 OF 6 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 CONV-REQ         7.958                   *TYPE=CREATE LOC=DB1D
================================================================================

CONVERSATION REQUEST

                  CONVERSATION ID: X'031440B8'

                  RETURN CODE:     4
```

Figure 104. DTRAC Conversation Request Pop-Up Display

This pop-up display shows the following:

CONVERSATION ID
: Four-byte identifier for the conversation in hexadecimal.

RETURN CODE
: Return code from conversation manager.

Deadlock Pop-Up (IFCID 172)

Additional data for the DEADLOCK event is displayed in the Deadlock pop-up display, as shown in Figure 105.

```
BMC SOFTWARE --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    10:13:34  INTVL=> 5  LOG=> N  TGT==> DB2KSR
PARM ==> DET71,SEQ=2,LEVEL=2                          ROW 1 OF 41 SCROLL=> CSR

START: 10:09:33 AUTH: BOLSMR3  PLAN: RXDB2    CORR: BOLSMR3      CONN: DB2CALL
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
  DEADLOCK       10.375                   *RESOURCES INVOLVED=2
==============================================================================
   INTERVAL COUNTER: 2,075    TIME DETECTED: 2001-05-09-10.09.44.043282
-----------------------------------------------------------------------------
             LOCK TYPE:      ROW
             RESOURCE NAME:  DB=DSN8D71A   OB=DSN8S71P
             RESOURCE #:     X'00000201'
             HASH VALUE:     X'08041302'
             FUNCTION:       LOCK

                 ---- HOLDER ----                ---- WAITER ----
 PLAN:           RXDB2                           RXDB2
 CORRELATION ID: BOLSMR4                         BOLSMR3
 CONNECTION ID:  DB2CALL                         DB2CALL
 OWNING WU:      003900A7-0F14A708               00390088-0F1488F8
 DURATION:       COMMIT                          MANUAL
 MEMBER:         DB2K                            DB2K
 STATE:          EXCLUSIVE                       SHARED
 CACHED STMT ID: 00000000                        00000000
 WORTH VALUE:                                    X'11'
    -- LUW ID:   NET ID   LUNAME   UNIQUE-VALUE CC   TOKEN --
       HOLDER:   USBMCN01 LUDB2K   B5CF17B86CE9 0001       40
       WAITER:   USBMCN01 LUDB2K   B5CF17B95DD7 0001       41
-----------------------------------------------------------------------------
             LOCK TYPE:      DATA PAGE
             RESOURCE NAME:  DB=DSN8D71A   OB=DSN8S71E
             RESOURCE #:     X'00001200'
             HASH VALUE:     X'00018C12'
             FUNCTION:       LOCK

                 ---- HOLDER ----                ---- WAITER ----
 PLAN:           RXDB2                           RXDB2
 CORRELATION ID: BOLSMR3                         BOLSMR4
 CONNECTION ID:  DB2CALL                         DB2CALL
 OWNING WU:      00390088-0F1488F8               003900A7-0F14A708
```

Figure 105. DTRAC Deadlock Pop-Up Display

This pop-up display shows the following:

INTERVAL COUNTER
: Deadlock interval counter.

TIME DETECTED
: Time (STCK) the deadlock was detected.

The following information is shown for each resource involved in the deadlock.

LOCK TYPE
: Type of locked resource, such as a data page, row, or page set in a table space or index, or an internal lock to control special processing such as utility serialization, data set open, or altering a buffer pool. See the “Lock Type Table” appendix for a complete list. The long version of the lock type is shown in this pop-up.

RESOURCE NAME
: Identifies the specific locked resource. The most common values are DB=database OB-pageset, but there are other possibilities, such as buffer pool ID. See the “Lock Type Table” appendix for a complete list.

RESOURCE #
: Sometimes referred to as the *small resource name*. It consists of a hex number. For example, for a DATA PAGE lock, a physical page number is shown as X‘hhhhhh00’. An INDEX PAGE lock on the root page is indicated with the special value of X‘000002FF’. A ROW lock is shown as X‘hhhhhhrr’, where the first six digits are the page number and the last two are the RID. For some resources, these numbers are blank. See the “Lock Type Table” appendix for a complete list.

HASH VALUE
: Hash value of the locked resource name.

FUNCTION
: Requested function by the thread waiting for the resource, which can be

  LOCK function.
  UNLOCK function.
  CHANGE function.

HOLDER/WAITER
: Identification of both the holder and the waiter for this resource is shown in this section.

PLAN
: Plan name of the thread holding the resource.
  Plan name of the thread waiting for the resource.

CORRELATION ID
: Correlation ID of the thread holding the resource.
  Correlation ID of the thread waiting for the resource.

CONNECTION ID
: Connection ID of the thread holding the resource.
  Connection ID of the thread waiting for the resource.

OWNING WU
: Holder’s owning work unit.
  Waiter’s owning work unit.

DURATION
Lock duration of the thread holding the resource and the lock duration requested by the thread waiting for the resource, which can be one of the following:

MANUAL
Lock is held only as long as necessary.

MANUAL+1
Temporary change of consistency level from CS to RR during BIND and DDL (RDS changes back internally).

COMMIT
Lock is held until COMMIT.

COMMIT-HOLD
Locks needed to maintain the position of a cursor opened *with hold* are held past COMMIT.

ALLOCATION
Lock is held until deallocation.

PLAN
Lock duration defined when the plan is bound.

FREE ALL LOCKS
Duration to free all locks.

MEMBER
Name of the DB2 member of a data sharing group (DB2 4.1 or higher) for the holder and the waiter. If the member name is different, this deadlock involved global contention.

STATE
Lock state of the thread holding the resource and the lock state requested by thread waiting for the resource, which can be

INTENT SHARE
INTENT EXCLUSIVE
SHARED
UPDATE
SHARED INTENT EXCLUSIVE
EXCLUSIVE

CACHED STMT ID
SQL statement unique identifier. This ID is a generated number for uniquely identifying a statement in the prepared statement cache.

WORTH VALUE
Waiter's DB2 assigned worth value. DB2 uses this value to determine the deadlock victim.

LUW ID
Logical unit of work ID of the thread holding the resource.
Logical unit of work ID of the thread waiting for the resource.

Note: This field is not valid if the thread is not a distributed thread. For non-distributed threads, it contains an asterisk. For allied distributed and DBAT threads, it contains a logical unit of work ID.

Distributed SQL Pop-Up (IFCID 168)

Additional data for the DIST-SQL event is displayed in the Distributed SQL pop-up display, as shown in Figure 106.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:17:53  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=2                                ROW 1 OF 5 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 DIST-SQL         2.320                   *SELECT * FROM DB1D.SYSIBM.SYSCOLUMN+
================================================================================

DISTRIBUTED SQL STATEMENT

SELECT * FROM DB1D.SYSIBM.SYSCOLUMNS ORDER BY NAME
```

Figure 106. DTRAC Distributed SQL Pop-Up Display

This pop-up display shows the SQL statement text.

Drain Pop-Up (IFCID 212)

Additional data for the DRAIN event is displayed in the Drain pop-up display, as shown in Figure 107.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC            INPUT   08:57:47  INTVL=> 5  LOG=> N  TGT==> DB2F
PARM ==> DRAINS,SEQ=3,LEVEL=3                         ROW 1 OF 5 SCROLL=> CSR

START: 08:53:56 AUTH: CJN1     PLAN: DSNUTIL  CORR: CJN1LD       CONN: UTILITY
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 DRAIN            7.995                   *DB=CJN1DB1  OB=CJN1AUS  PART=0
==============================================================================
             REQUEST TYPE:   DRAIN
             CLASS:          WRITE
             LOCK MODE:      EXCLUSIVE LOCK
             RETURN CODE:       0
             FAIL REASON:    UNKNOWN
```

Figure 107. DTRAC Drain Pop-Up Display

This pop-up display shows the following:

REQUEST TYPE
Request type, which can be one of the following:

DRAIN
RELEASE THE DRAIN

CLASS
Drain class, which can be one of the following:

CURSOR STABILITY READ
REPEATABLE READ
WRITE

LOCK MODE
Mode of the drain lock requested, which can be one of the following:

INTENT EXCLUSIVE
EXCLUSIVE LOCK
DRAIN IS RELEASED

RETURN CODE
Result of the drain request.

FAIL REASON

Reason for the unsuccessful drain (if not SUCCESSFUL), which can be one of the following:

- Unknown
- Deadlock
- Timeout
- IRLM is out of storage
- Error in IRLM
- Timeout waiting for a zero claims count

Drain Wait Pop-Up (IFCID 213/214)

Additional data for the DRAIN-WAIT event is displayed in the Drain Wait pop-up display, as shown in Figure 108.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    11:59:34  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> SMRRUN3,SEQ=7,LEVEL=3                        ROW 1 OF 6 SCROLL=> CSR

START: 11:48:06 AUTH: BOLSMR4  PLAN: DSNUTIL  CORR: BOLSMR2L     CONN: UTILITY
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 DRAIN-WAIT       1.830 4,982 us   342 us *DB=DB2SMR2  OB=TS00003  PART=0
================================================================================
             CLASS:          WRITE DRAIN
             FUNCTION:       LOCK
             STATE:          EXCLUSIVE
             DURATION:       ALLOCATION
             SUSPEND REASON: INTER-SYSTEM COMM
             RESUME REASON:  NORMAL
             GLOBAL REASON:  IRLM GLOBAL RESOURCE CONTENTION
```

Figure 108. DTRAC Drain Wait Pop-Up Display

This pop-up display shows the following:

CLASS
Type of lock resource class, which can be one of the following:

CURSOR STABILITY DRAIN
REPEATABLE READ DRAIN
WRITE DRAIN

FUNCTION
IRLM function code, which can be one of the following:

LOCK function.
UNLOCK function.
CHANGE function.

STATE
Lock state, which can be one of the following:

ANY (CHANGE or UNLOCK can use this)
INTENTED SHARE
INTENTED EXCLUSIVE
SHARED
UPDATE
SHARED INTENTED EXCLUSIVE
EXCLUSIVE

DURATION
LOCK duration, which can be one of the following:

MANUAL
The lock is held only as long as necessary.

MANUAL+1
Temporary change of consistency level from CS to RR during BIND and DDL (RDS changes back internally).

COMMIT
The lock is held until COMMIT.

COMMIT+1
Locks needed to maintain the position of a cursor opened *with hold* are held past COMMIT.

ALLOCATION
The lock is held until deallocation.

PLAN
Lock duration defined when the plan is bound.

FREE ALL LOCKS
The duration to free all locks.

SUSPEND REASON
Reason for the suspend, which can be one of the following:

IRLM LATCH CONTENTION
LOCAL RESOURCE CONTENTION
GLOBAL RESOURCE CONTENTION
INTER-SYSTEM COMMUNICATION

RESUME REASON
Reason for the resume, which can be one of the following:

DEADLOCK
TIMEOUT
NORMAL

GLOBAL REASON
Reason for data sharing contention, which can be one of the following:

MVS XES RESOURCE CONTENTION
IRLM GLOBAL RESOURCE CONTENTION

EXPLAIN Pop-Up (IFCID 022)

Additional data for the EXPLAIN event is displayed in the EXPLAIN pop-up display, as shown in Figure 109.

Note: Some fields are not displayed if the data is not relevant for the version of DB2 you are running.

```
BMC SOFTWARE --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==>DTRAC INPUT  11:57:59 INTVL=>5LOG=>NTGT==>DB2K
PARM ==>JOINT,SEQ=3,LEVEL=2                        ROW 1 OF 76SCROLL=>CSR
EXPAND: CATALOG                                         HIST TGT--- DB2J
START: 09:51:44 AUTH: BOLCJN2  PLAN: DSNESPRR CORR: BOLCJN2      CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 EXPLAIN          2.690                   *PLAN=DSNESPRR COST(5,635 K)
==============================================================================
QUERY NUMBER: 2             EXPLAIN DATE..: 2001-04-23 09:51:2260
GROUP MEMBER: DB2J          STATEMENT TYPE: SELECT   PARENT QRY BLK#: 0
PROGRAM NAME: DSNESM68     COLLECTION ID.: DSNESPRR
VERSION NAME:
WHEN_OPTIMIZ: AT BIND TIME USING DEFAULT VALUES
OPTIMIZATION HINT ID:
PROCESSOR COST IN MILLISECONDS:                 IN SU:
COST CATEGORY:               CATEGORY B REASON:
------------------------------------------------------------------------------
BLKNO:   1        SEQNO:   1    MXSEQNO:   0
METHOD: FIRST TABLE
ACCESS: INDEX SCAN                          PREFETCH: NONE
 COL_FN:
 PAGE RANGE SCAN:  NO
TSLOCK: S     TABLE TYPE: TABLE                 STARJOIN BY DATA MGR
TABLE CARDINALITY: 4,774.0    PAGES: 72
TABLE: KELI01  .DEPT01                  CORRELATION: D1
INDEX:                                  MATCHING:
PARALLELISM: CPU                        RLF: NONE DISABLED
          ACCESS DEGREE:     4          ACCESS GROUP ID:   1
------------------------------------------------------------------------------
BLKNO:   1        SEQNO:   1    MXSEQNO:   1
METHOD: FIRST TABLE
ACCESS: MULTINDEX/UNION        INDEX ONLY    PREFETCH: NONE
 COL_FN:
 PAGE RANGE SCAN:  NO
TSLOCK: S
TABLE CARDINALITY: 4,774.0    PAGES: 72
TABLE: KELI01  .DEPT01                  CORRELATION: D1
INDEX: KELI01  .XDEPT01                 MATCHING: 0         KEYS
PARALLELISM: CPU                        RLF: NONE DISABLED
          ACCESS DEGREE:     4          ACCESS GROUP ID:   1
------------------------------------------------------------------------------
```

Figure 109. DTRAC EXPLAIN Pop-Up Display

Expand: From the DTRAC EXPLAIN pop-up display you can expand to

CATALOG
: The corresponding RxD2 catalog display for the table defined in the TABLE line, if RxD2 is installed. From the table display, you can access information on columns, indexes, keys, referential constraints, or plan dependencies. You can also see the statistics that are used in access path selection and the timestamps of when these values were updated. (See the IBM *DB2 Administration Guide* for a complete description.)

Note: Access is possible to remote DB2 subsystems only if connected with DDF to a local DB2 (in the same MVS system as the TS).

Description: Provides critical access path selection information similar to EXPLAIN. This information is produced by the DB2 optimizer when it binds SQL statements. A cost factor is also provided.

The EXPLAIN pop-up shows the access path chosen and can be used to evaluate SQL coding techniques and identify potential performance problems. Detail trace captures the information for the following:

- Dynamic SQL at execution, including:
  - QMF queries
  - SPUFI requests
  - Dynamic SQL within static plans
- Static SQL at BIND, REBIND, or EXPLAIN

Note: This information is captured whether or not EXPLAIN is specified on the BIND.

An EXPLAIN event always is preceded by a BIND-TEXT event that contains the full SQL statement text in its pop-up. You can use PF10/11 to move back and forth between the BIND-TEXT and EXPLAIN pop-ups. For multiple statements in a plan, the EXPLAIN entries are interleaved with the BIND-TEXT entries. This compares to the QUERYNO level in the PLAN_TABLE. There can be multiple EXPLAIN pop-ups for a statement in those cases where there are multiple QBLOCKNO entries in the PLAN_TABLE.

Note: Detail trace collects the access path information provided by the DB2 optimizer in the IFCID 22 trace record (see "Event Data" on page 259). These fields are equivalent to the columns available in a PLAN_TABLE and the column names are referenced in the following field descriptions (see the IBM *DB2 Administration Guide*). The following differences to the columns in the PLAN_TABLE exist for different releases of DB2:

- Not available:
  - REMARKS
- Not available for DB2 6.1 and below:
  - PROCESSOR COST IN MILLISENONDS
  - IN SU
  - COST CATEGORY
  - CATEGORY B REASON
  - TABLE CARDINALITY
  - PAGES
- Not available for DB2 4.1 and below:
  - MIXOPSEQ
  - COL_FN_EVAL
  - PROGRAM NAME
  - VERSION
  - WHEN OPTIMIZED
  - STATEMENT TYPE
- Not available for DB2 3.1:
  - TSLOCKMODE
  - CREATOR
  - ACCESSCREATOR
  - TIMESTAMP
- ACCESSTYPE of blank
  - Does not generate an EXPLAIN trace event

Header Fields:

The header fields are valid for all blocks so are shown once at the top of the display when available.

QUERY NUMBER
Number that identifies the statement being explained. PLAN_TABLE shows an increment to this value for each BIND-TEXT/EXPLAIN pair in the trace.
(QUERYNO in PLAN_TABLE) (DB2 4.1 and later)

GROUP MEMBER
Member name of this DB2 if in a data sharing group, otherwise blank.
(GROUP-MEMBER in PLAN_TABLE) (DB2 4.1 and later)

PROGRAM NAME
Name of the program or package containing the statement being explained.
(PROGNAME in PLAN_TABLE) (DB2 5.1 and later)

This field applies only to embedded EXPLAIN statements and to statements explained as the result of binding a plan or package. It is blank if not applicable.

VERSION NAME
Version ID for the package.
(VERSION in PLAN_TABLE) (DB2 5.1 and later)

This field applies only to an embedded EXPLAIN statement executed from a package or to a statement that is explained when binding a package. It is blank if not applicable.

WHEN_OPTIMIZ
Indicates when DB2 determines the access path.
(WHEN_OPTIMIZE in PLAN_TABLE) (DB2 5.1 and later)

Possible values are as follows:

AT BIND TIME USING DEFAULT VALUES
Access path is determined at bind time using default values.

BIND TIME BUT REOPTIMIZED AT EXEC WITH INP VARS
Access path is determined at bind time using default values but is determined again at run time using the values in input variables.

EXECUTION TIME USING VALUES OF INPUT VARS
Access path is determined at run time using the values in input variables.

OPTIMIZATION HINT ID
Value of optimization hints (OPTHINTS). This field contains the value of bind option opthints or the value of special register current optimization hints.
(OPTHINT in PLAN_TABLE) (DB2 6.1 and later)

PROCESSOR COST IN MILLISECONDS
Estimated processor cost for the SQL statement, in milliseconds.
(DB2 6.1 and later)

IN SU
Estimated processor cost for the SQL statement, in service units.
(DB2 6.1 and later)

COST CATEGORY
Specifies whether the cost category for the statement is A or B.
(DB2 6.1 and later)

CATEGORY B REASON
Specifies the reason code for cost category B.
(DB2 6.1 and later)

Possible values are as follows:

– THERE ARE HOST VARIABLES

  There are host variables parameter markers or special registers in range or between predicates.

– TABLE CARD IS MISSING

  The table cardinality is missing for one or more tables.

– TRIGGERS MAY BE USED

  There are insert, update, or delete triggers defined on the target table.

– UDFS MIGHT BE INVOKED

  There are user-defined functions referenced in the SQL statement.

– REFERENTIAL CONSTRAINTS

  A table that is the target of a delete function has referential constraints defined on it.

– HAVING CLAUSE

  A HAVING clause causes an SQL statement to be assigned to COST CATEGORY B.
  (DB2 7.1 and later)

EXPLAIN DATE/TIMESTAMP
Date and time the statement was bound.
(TIMESTAMP in PLAN_TABLE)

STATEMENT TYPE
Type of statement.
(QBLOCK_TYPE in PLAN_TABLE) (DB2 5.1 and later)

Possible values are as follows:

- SELECT - SELECT
- INSERT - INSERT
- UPDATE - UPDATE
- DELETE - DELETE
- SELUPD - SELECT with FOR UPDATE OF
- DELCUR - DELETE WHERE CURRENT OF CURSOR
- UPDCUR - UPDATE WHERE CURRENT OF CURSOR
- CORSUB - Correlated subquery
- NCOSUB - Noncorrelated subquery

COLLECTION ID
Collection ID for the package, if applicable.
(COLLID in PLAN_TABLE) (DB2 4.1 and later)

PARENT QRY BLK#
Parent Query Block number.
(PARENT_QBLOCKNO in PLAN_TABLE) (DB2 7.1 and later)

Sequence Fields:

The next three fields are used to show the sequence of the access path steps.

BLKNO
Query block number shows the position of the query in the statement being explained.
(QBLOCKNO in PLAN_TABLE)
(DB2 4.1 and later, blank for earlier releases)

SEQNO
Step number for the execution of the query block.
(PLANNO in PLAN_TABLE)

MXSEQNO
Sequence number of a step in a multiple index operation.
(MIXOPSEQ in PLAN_TABLE) (DB2 5.1 and later)

METHOD

The fields on this line shows information on the method used in this step. This is either a join type or an indication that this step accesses the first table or is doing an additional sort. This information should be used in conjunction with that in the ACCESS line.

METHOD
: Join method used.
(METHOD in PLAN_TABLE)

FIRST TABLE
: First table accessed, or continuation of previous table accessed, or not used.

MERGE SCAN JOIN
: Merge scan join. The present composite table and the new table are scanned in the order of the join columns, and matching rows are joined.

A merge scan join is usually selected when two large tables are joined and one or both tables do not have a good index. Any table without a good index on the join columns will be sorted (see SORT line).

NESTED LOOP JOIN
: Nested loop join. For each row of the present composite table, matching rows of a new table are found and joined.

A nested loop join is often selected by DB2 when good indexes exist on the joined columns and one or both tables involved in the join are data clustered in join sequence. Also, nested loop joins tend to get selected when a small number of indexed rows in one table are matched with a larger set of rows from another table. Nested loop joins may invoke sequential detection to activate prefetch to reduce the amount of I/O.

ADDITIONAL SORT
: Additional sort required by ORDER BY, GROUP BY, SELECT, DISTINCT, UNION, or a quantified predicate. This step does not access a new table.

HYBRID JOIN
: Hybrid join. The current composite table is scanned in the order of the join-column rows of the new table. The new table is accessed using list prefetch.

A hybrid join is a mix of nested loop and merge scan join processes. It is invoked when there are non-unique indexes of the join predicate that either may have many duplicate values or are not in clustering data sequence. At least one of the tables must have an index on the join criteria. Check the SORT NEW TABLE to see if a sort is required for the join. A clustered index bypasses the need for a sort.

LEFT OUTER JOIN or FULL OUTER JOIN

Outer join type, if applicable. Right outer join converts to a left outer join when used. All other joins are inner joins.
(JOIN_TYPE in PLAN_TABLE) (DB2 4.1 and later)

Outer joins are all tests for non-existence relationships. Inner joins attempt to find matches of rows from one table to another. Outer joins additionally attempt to find rows in one table that have no corresponding row in another table. LEFT and RIGHT outer joins simply describe the order in which the tables are searched for *missing rows*. A FULL outer join attempts to find all matching rows and all non-matching rows from both tables.

MERGE COLS: nn

Number of columns joined during a merge scan join, if applicable.
(MERGE_JOIN_COLS in PLAN_TABLE) (DB2 4.1 and later)

ACCESS

One or two lines show information on the method used to access the new table (the table being read).

ACCESS
Method used to access the new table.
(ACCESSTYPE plus INDEXONLY in PLAN_TABLE)

Access methods are as follows:

INDEX SCAN
Scan of the index identified in the INDEX line.

In general, for random access to a table of any size, it is desirable to use an index. If none is used, check for the existence of an index with usable columns and that RUNSTATS exist for the objects so that DB2 can make informed decisions. See the INDEX line for the number of matching columns, or whether this is a nonmatching index scan.

SEQUENTIAL SCAN
Table space sequential scan. In most cases, sequential prefetch will be invoked.

It is most often used for one of these reasons:

- No index is available, or there are no predicates that match the index columns
- High percentage of the rows in the table is returned
- Matching indexes have low cluster ratios and are not efficient for large amounts of data.

INDEX ONLY
Scan of the index is sufficient to satisfy the request. No data pages need to be accessed. This typically occurs with an existence check.

Note: For DB2 3.1, this is not always identifiable in the IFCID 22, although it is for DB2 4.1 and later. It may sometimes show an INDEX SCAN when an INDEX ONLY SCAN is actually used.

INDEX SCAN (SQL ‘IN’)
Special form of matching index scan is an SQL ‘IN-list’ scan, using the index identified in the INDEX line.

This creates a series of matching index scans with the values in the IN predicate being used for each scan.

INDEX SCAN (ONE FETCH)
One fetch index access requires only one row to be retrieved.

This path is selected only when there is one table in the query and the MIN function is used for an ascending index column, or the MAX function is used for a descending index column. The index provides the proper ordering so that only one qualified row is required to give the result of the column function. It is the best possible access path and is chosen whenever it is available.

MULTINDEX
Multiple index access.

Multiple index accesses are scans of more than one index to satisfy an SQL query. It is a good access path when no single index matches a query but a combination of indexes provide relatively efficient access. From a performance perspective, it will never be as good as a direct match with a single index, but when there are many queries that need different columns, using multiple indexes is the next best choice.

Each of the indexes is scanned for qualifying record IDs (RIDs) and then merged (an intersection of RIDs common to both indexes if AND is specified in the SQL, or a union of all RIDs if OR is specified). List prefetch is used with the resulting RID list. RID pool failures at this point would result in a table space scan.

Four multiple index access flags are shown (DB2 5.1 and later):

START — Table is accessed by a multiple index scan, followed by SCAN, INTERSECT, or UNION (described below).

SCAN — Index scan of the index identified in the INDEX line.

INTERSECT — Intersection of multiple indexes.

UNION — Union of multiple indexes.

PREFETCH: x
Type of prefetch to be used.
(PREFETCH in PLAN_TABLE)

The possible values are as follows:

- SEQ for sequential prefetch
- LIST for list prefetch
- NONE if prefetch was not chosen at bind time

Prefetch can substantially improve performance and reduce processor and I/O costs when large amounts of data are being accessed. For a table space scan, sequential prefetch is almost always used. OPTIMIZE FOR n ROWS can be used to prevent it. Sequential prefetch may also be used for index scans.

Using sequential detection techniques, DB2 may sometimes be able to use sequential prefetch at runtime (called dynamic prefetch) even when it is not shown in the EXPLAIN.

COL_FN:
Indicates when an SQL column function is evaluated.
(COLUMN_FN_EVAL in PLAN_TABLE) (DB2 5.1 and later)

Possible values are as follows:

R — Column function is evaluated during data retrieval.

S — Column function is evaluated during sort.

blank — Column function is evaluated at execution.

PAGE RANGE SCAN:

Page range scan is shown below ACCESS, if applicable. It applies to table spaces scans or non-partitioning index scans. It allows only the partitions needed to be scanned. (PAGE_RANGE in PLAN_TABLE) (DB2 4.1 and later)

DB2 takes full advantage of partitioned table spaces by doing partition scans or page range scans. If a matching predicate exists on a clustering index, DB2 can scan just a subset of the partitions or a range of pages smaller than the entire table space. The reduces both I/O and processor time.

TSLOCK

Table space or table lock mode is shown in this line. (Individual page or row locks will also be taken and released as indicated in the BIND parameters and LOCKSIZE.) In addition, the table type and whether or not it is processed using Star Join is shown.

TSLOCK

Lock mode of the table space that contains the table if the table space is not segmented, or if the LOCKSIZE is TABLESPACE. Otherwise, the lock mode of the table. (TSLOCKMODE in PLAN_TABLE) (DB2 4.1 and later)

Possible values are as follows:

- IS - intent share
- IX - intent exclusive
- S - share
- U - update
- X - exclusive
- SIX - share with intent exclusive
- N - UR isolation
- NS - no isolation level determined/share if CS/RR, none if UR
- NIS - no isolation level determined/intent share if CS/RR, none if UR
- NSS - no isolation level determined/intent share if CS, share if RR, none if UR
- SS - no isolation level determined/intent share if CS/UR, share if RR

TABLE TYPE

Type of table.
(TABLE_TYPE in PLAN_TABLE) (DB2 7.1 and later)

Possible values are as follows:

- TABLE
- FUNCTION - TABLE FUNCTION
- WORKFILE
- QUEUE - TABLE QUEUE (NOT MATERIALIZED)

STARJOIN BY DATA MGR

The table is processed by DB2 Data Manager using the Star Join technique.
(DB2 6.1 and later)

TABLE CARDINALITY

This line shows information about table cardinality.

TABLE CARDINALITY
: Table cardinality in floating point.
(DB2 6.1 and later)

PAGES
: Number of pages for the table.
(DB2 6.1 and later)

DIRECT ACCESS POSSIBLE
: Message displayed if DB2 can use direct access to a table row without a table space or index scan.
(PRIMARY_ACCESSTYPE in PLAN_TABLE) (DB2 6.1 and later)

TABLE

This line shows information about the table being accessed.

TABLE
: Table name being accessed, or TEMPORARY for a DB2 internal result table. For an outer join, you may see DSNWFQBxx for a work file.
(TNAME in PLAN_TABLE)

CORR: x
: Where x is the table qualifier (correlation name) specified in the statement for this table, if present.
(CORRELATION_NAME in PLAN_TABLE) (DB2 4.1 and later)

CREATOR: x
: Where x is the table creator name if present.
(CREATOR in PLAN_TABLE) (DB2 4.1 and later)

INDEX

This line shows information about the index used and the number of matching columns.

The index or indexes chosen are important because DB2 determines how it will retrieve the data based upon how closely the SQL WHERE clause and ORDER BY clause match an existing index.

RUNSTATS that reflect reasonably closely actual table and index contents are needed to ensure the best possible decisions can be made.

INDEX
: Index name being used, or NONE.
(ACCESSNAME in PLAN_TABLE)

MATCHING nn INDEX KEYS
: Identifies how many key fields are used for match processing during an INDEX SCAN of the table space; where nn can be 1 to 99.
(MATCHCOLS in PLAN_TABLE)

A matching index scan is an index scan that traverses directly through the index structure to the data rows being sought. DB2 evaluates multi-columned indexes from left to right, that is, from the first column specified to the second, third, and so on. Generally speaking, the higher the number of columns matched in a matching index scan, the more efficient the query is.

If possible, include all columns of an existing index in the WHERE clause to encourage DB2 to select a matching index scan. An exception is when one is a range predicate (greater or less-than criteria). Matching is not done past the range predicate. However, predicates that are not part of the matching columns can still be used for index screening to reduce the number of qualifying rows during the index scan.

NONMATCHING INDEX
: All keys in the index must be examined.
(MATCHCOLS=0 in PLAN_TABLE)

A nonmatching index scan is an index scan of all the index leaf pages. DB2 chooses this access type when it determines that the data is in the index, but that it must read all leaf pages (that contain the pointers to the data) to identify and extract the qualifying rows. This usually occurs when the query specifies columns in the WHERE clause that are not the high order columns of the index (for example, the fourth and fifth columns instead of the first and second).

It is efficient only in a few cases; for example, when index screening is used; sometimes when OPTIMIZE FOR n ROWS is specified; or when there is more than one table in a nonsegmented table space.

DB2 chooses this path over a table space scan if the table and the index are large. If small, a table space scan may be more efficient. It is a good idea to include as many predicates in the WHERE clause as possible, since DB2 can screen more non-qualifying rows during the index scan.

CREATOR: x
: Where x is the index creator name if present.
(ACCESSCREATOR in PLAN_TABLE) (DB2 4.1 and later)
(Shown as part of the index name in DB2 5.1 and later)

SORT

This line shows sort information for either a new table or a composite table.

DB2 sorts use work files in DSNDB07 to perform the sort. If a large number of records are sorted, the query may not be as efficient as expected. Also, sorting unneeded columns increases the sort record size and use of work file resources. Possible solutions for sort problems are to create indexes to avoid the need for sort (since index keys are always kept sorted), or in some cases changing the application to use an external sort using standard sort utilities.

DB2 also sorts RIDs into ascending page number order to perform list prefetch. This sort is very fast and is done completely in memory. RID sorts are usually performed whenever list prefetch is used, but this is not indicated in the EXPLAIN information, except for a RID list sort on the inner table for a hybrid join when no highly clustered index is available.

Be aware that data is sorted when the cursor is opened. If a cursor that requires a sort for ORDER BY is closed and then reopened, the sort must be performed again, causing performance degradation.

SORT
: Table being sorted.

NEW TABLE
: Indicates a new table sort (sometimes called the inner table).

The only reason DB2 sorts the new table is for join processing. (SORTN_xxx in PLAN_TABLE)

COMPOSITE TABLE
: Indicates a composite table sort (an outer table or results table).

For hybrid join and nested loop join, the composite table can be sorted to make the join more efficient. For a merge scan join, both the composite table and new table need to be sorted unless an index can be used. If multiple reasons for the sort are shown, there is still only one sort performed; for example, ORDERBY and UNIQUE would mean that the single sort puts the rows in order and removes any duplicate rows as well. These two reasons are seen for a sort for a non-correlated subquery. (SORTC_xxx in PLAN_TABLE)

Reason for sort
: Possible reasons are as follows:

JOIN
: Indicates a sort is required for the JOIN operation.

ORDER BY
: Indicates a sort for an ORDER BY clause or a quantified predicate in the SQL statement.

GROUP BY
: Indicates a sort for a GROUP BY clause in the SQL statement.

UNIQUE
: Indicates a sort for a DISTINCT predicate in the SQL statement or during UNION processing to eliminate duplicates.

UNIQUE (COL FUNC)
: Indicates a sort for a DISTINCT predicate for a column function in the SQL statement, such as COUNT(DISTINCT col1).

PARALLELISM

If parallelism is used, this section shows the applicable information.

If no parallelism is used, the section is not shown.

PARALLELISM
Kind of parallelism used, if applicable. This can be either I/O, CPU (DB2 4.1 and later), or sysplex (DB2 5.1 and later).
(PARALLELISM_MODE in PLAN_TABLE)

Parallelism is considered by DB2 only if DEGREE(ANY) is specified at BIND/REBIND for static SQL, or SET CURRENT DEGREE sets the special register to 'ANY' for dynamic SQL. A degree of '1' turns off parallelism.

DB2 decides whether to use parallelism based on whether the calculated I/O or processor cost is high. At runtime, a parallel query can be reduced to fewer concurrent I/O streams or fall back to a sequential scan if insufficient buffer pool resources are available (see VPPSEQT, the parallel sequential steal threshold). There are several other fallback reasons, such as use of an updateable cursor, or the lack of the ESA sort feature or MVS/ESA 5.2 enclave services.

RLF
Defines Resource Limit Facility (RLF) control of parallelism for dynamic SQL only. When disabled, the query reverts to a table space scan.
(not in PLAN_TABLE) (DB2 4.1 and later)

Possible values can be as follows:

NONE DISABLED
No query parallelism is disabled. The query will run as planned, based only on the availability of sufficient buffers for parallel processing.

I/O DISABLED
I/O query parallelism is disabled.

CPU DISABLED
CPU query parallelism is disabled.

I/O AND CPU DISABLED
I/O and CPU query parallelism are disabled.

MULTI-CEC DISABLED
Sysplex query parallelism is disabled.

MULTI-CEC AND I/O DISABLED
Sysplex query parallelism and I/O parallelism are disabled.

MULTI-CEC AND CPU DISABLED
Sysplex query parallelism and CPU parallelism are disabled.

ALL DISABLED
Sysplex query parallelism, I/O parallelism, and CPU parallelism are disabled.

RLF NOT USED
RLF does not affect this statement.

N/A
Not applicable for DB2 3.1.

ACCESS DEGREE x

Degree of parallel processing (number of parallel I/O streams) to access the new table (must be partitioned).
(ACCESS_DEGREE in PLAN_TABLE)

The degree of parallel processing can be displayed as:

blank	None
0	Degree determined at run time
n	Planned degree

ACCESS GROUP ID xxxx

The parallel group ID for accessing the new table.
(ACCESS_PGROUP_ID in PLAN_TABLE)

JOIN DEGREE x

Degree of parallel processing (number of parallel I/O streams) to join the new table with composite intermediate results.
(JOIN_DEGREE in PLAN_TABLE)

The degree of parallel processing can be displayed as:

blank	None
0	Degree determined at run time
n	Planned degree

JOIN GROUP ID xxxx

Parallel group ID for joining the new table.
(JOIN_PGROUP_ID in PLAN_TABLE)

SORT NEW TABLE GROUP ID xxxx

Parallel group ID for sorting the new table when DB2 is able to logically partition the work files.
(SORTN_PGROUP_ID in PLAN_TABLE) (DB2 4.1 and later)

SORT COMPOSITE GROUP ID xxxx

Parallel group ID for sorting a composite table when DB2 is able to logically partition the intermediate results composite table.
(SORTC_PGROUP_ID in PLAN_TABLE) (DB2 4.1 and later)

Free Pop-Up (IFCID 110/111)

Additional data for the FREE event is displayed in the Free pop-up display, as shown in Figure 110.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:13:40  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> D1,SEQ=000001,LEVEL=2                ROW     1 OF     9 SCROLL=> CSR

START: 10:12:47 AUTH: CIR4X    PLAN: BINDCT   CORR: CIR4X        CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 FREE             2.294   451 ms    46 ms *RC=(   0) PACKAGE=DB2CPKG1
==============================================================================
        OWNER:           CIR2
        OBJECT TYPE:     PACKAGE
        QUALIFIER:       DBADB2X
        PKG LOCATION:    DB2D
        COLLECTION-ID:   BBXCOL3
        PACKAGE-ID:      SAJUYH2I
        TOKEN:           X'D10562E50452CD32'
        VERSION:
```

Figure 110. DTRAC Free Pop-Up Display

This pop-up display appears only if a package is being freed. It shows the following:

OBJECT TYPE
Type of object bound, either plan or package.

QUALIFIER
Qualifier for unqualified objects.

PKG LOCATION
Package location. This is blank if the local location is unnamed.

COLLECTION-ID
Collection ID of the package.

PACKAGE-ID
Package ID of the package.

TOKEN
Time-stamp for the BIND.

VERSION
Version identification name.

Hybrid Join Failure Pop-Up (IFCID 190)

Additional data for the HJOIN-FAIL event is displayed in the Hybrid Join Failure pop-up display, as shown in Figure 111.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    19:01:59  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=1,LEVEL=2                              ROW 1 OF 13 SCROLL=> CSR

START: 14:14:21 AUTH: CMRL     PLAN: LILYREP  CORR: CMRLJOB      CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
HJOIN-FAIL       15.409                   *#RIDS=999,999 FINAL RIDLIST NO STRG
================================================================================
CASE:   1
IMPACT: T - USE TABLESPACE SCAN TO FETCH INNER TABLE ROW
```

Figure 111. DTRAC Hybrid Join Failure Pop-Up Display

This pop-up display can show the following short explanatory phrases for the type of failure:

INTERM. TABLE IS FULL
RID LIST SORT NO STRG
MAX RIDS AT LIST APND
RID LIST APND NO STRG
FINAL RIDLIST NO STRG
FINAL RID THRESHOLD
SINGLE INDEX NO STRG
SINGLE INDEX THRESHOLD

Other fields shown include the following:

CASE — The hybrid join case, either 1 or 2.

IMPACT — The impact of the join failure. Possible values include:

- C — NO REPROCESS OF LAST OUTER ROW
- R — REPROCESS LAST OUTER ROW
- T — USE TABLESPACE SCAN TO FETCH INNER TABLE ROW

Lock Request Pop-Up (IFCID 021)

Additional data for the LOCK, UNLOCK, LOCK-CHG, and LOCK-QRY events is displayed in the Lock Request pop-up display, as shown in Figure 112.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    14:43:54  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> DATASHR,SEQ=4,LEVEL=3                       ROW 1 OF 18 SCROLL=> CSR
                                                              HIST TGT--- DB2H
START: 07:52:40 AUTH: BOLSMR4  PLAN: DSNTIA51 CORR: DB251REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
   LOCK           3.222              0 us *DB=DSNDB01  OB=DBD01    TYP=PAGE PLK
==============================================================================
             LOCK TYPE:     PAGE P-LOCK
             RESOURCE #:    X'0001000001'
             LOCK TOKEN:    07B072A4

             LOCK STATE:    X
             LOCK DURATION: INTEREST DURATION FOR P-LOCKS

        REQUEST TYPE/MODE:  UNCONDITIONAL
                            ACQUIRE
                            UNLOCK GROUP
                            RESULTANT STATE
                            MODIFY LOCK
                            PAGE LOCK
                            SYNCHRONOUS SLM REQUEST
                            LOCAL LOCK

          IRLM RETURN CODE: 00000000
                   SUBCODE: 00000000
```

Figure 112. DTRAC Lock Request Pop-Up Display

This pop-up display shows the following:

LOCK TYPE
: Type of locked resource, such as a data page, row, or page set in a table space or index, or an internal lock to control special processing such as utility serialization, data set open, or altering a buffer pool. See Appendix A, "Lock Type Table" on page 405 for a complete list. The long version of the lock type is shown in this pop-up.

RESOURCE #
: Sometimes referred to as the *small resource name*. It consists of a hex number. For example, for a DATA PAGE lock, a physical page number is shown as X'hhhhhh00'. An INDEX PAGE lock on the root page is indicated with the special value of X'000002FF'. A ROW lock is shown as X'hhhhhhrr', where the first six digits are the page number and the last two are the RID. For some resources, this field is blank. See Appendix A, "Lock Type Table" on page 405 for a complete list.

LOCK TOKEN
: IRLM lock request token

LOCK STATE
Requested state, which can be one of the following:

UNS	UNPROTECTED SHARE
IS	INTENT SHARE
IX	INTENT EXCLUSIVE
S	SHARED
U	UPDATE
SIX	SHARED INTENT EXCLUSIVE
NSU	NON-SHARED UPDATE
X	EXCLUSIVE

LOCK DURATION
MANUAL
TEMPORARY
COMMIT
ALLOCATION
PLAN
DURATION

Note: In LOCK DURATION, an added value of X‘41’ is described as COMMIT WITH HOLD CURSOR POSITION.

REQUEST TYPE/MODE
CONDITIONAL or UNCONDITIONAL
AUTOREL or ACQUIRE
SINGLE or GROUP UNLOCK
RESULTANT STATE or COUNT BY STATE
MODIFY LOCK or NON-MODIFY LOCK
PAGE LOCK or LOGICAL LOCK
ASYNCHRONOUS SLM REQUEST or SYNCHRONOUS REQUEST
GLOBAL LOCK or LOCAL LOCK

IRLM RETURN CODE
Return code from the IMS Resource Lock Manager

SUBCODE
Subcode returned from the IMS Resource Lock Manager

Lock Summary Pop-Up (IFCID 020)

Additional data for the LOCK-SUMMARY event is displayed in the Lock Summary pop-up display, as shown in Figure 113.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    10:20:29  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> LOCKS,SEQ=1,LEVEL=2                          ROW 1 OF 4 SCROLL=> CSR

START: 09:23:57 AUTH: BOLSMR4  PLAN: DSNUTIL  CORR: DB251REQ     CONN: UTILITY
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
  LOCK-SUMMARY    0.026                   *MAXPG(5) ESCL(0) TS(  4)
================================================================================
DATABASE TABLESPACE TS TYPE   MAX PAGE TABLE ESCL HIGHEST STATE LOWER TYPE
-------- ---------- --------- -------- ---------- ------------- ----------
DSNDB01  DBD01      SIMPLE           2                     IX         PAGE
DSNDB06  SYSDBASE   SIMPLE           1                     IX         PAGE
DEMOD51A DEMOS51P   SEGMENTED        0          0                     NONE
DSNDB01  SYSUTILX   SEGMENTED        1          0                     PAGE
```

Figure 113. DTRAC Lock Summary Pop-Up Display

This pop-up display shows the following row of statistics for each database:

DATABASE
: Database name

TABLESPACE
: Table space name

TS TYPE
: Table space type, which can be one of the following:

  SIMPLE
  SEGMENTED
  PARTITION
  PARTLOCK (partitioned table space that uses selective partition locking)

  (DB2 version 5 and later)

MAX PAGE
: Maximum number of page and row locks held concurrently in this table space

TABLE ESCL
: For segmented table spaces, the total number of tables that have escalated within the table space

HIGHEST STATE

For unsegmented table spaces, the highest page set lock state which can be one of the following:

ANY	ANY
UNS	UNPROTECTED SHARE
IS	INTENT SHARE
IX	INTENT EXCLUSIVE
S	SHARED
U	UPDATE
SIX	SHARED INTENT EXCLUSIVE
NSU	NON-SHARED UPDATE
X	EXCLUSIVE

LOWER TYPE

Type of lock for indexes, which can be one of the following:

NONE
PAGE
ROW
LOB

Lock Suspension Pop-Up (IFCID 044/045)

Additional data for the LOCK-SUSP event is displayed in the Lock Suspension pop-up display, as shown in Figure 114.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    11:50:31  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> LOCKS,SEQ=4,LEVEL=3                          ROW 1 OF 21 SCROLL=> CSR

START: 10:35:54 AUTH: BOLSMR4  PLAN: DSNTIA51 CORR: DB251REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 LOCK-SUSP        7.781 3,568 us   208 us *DB= DSNDB01 OB= DBD01   TYP=P/P CAST
==============================================================================
             LOCK TYPE:     PAGESET/PARTITION CASTOUT PLOK
             RESOURCE #:    X'00001F0000'

        PARENT LOCK TOKEN:  08233044
             HASH VALUE:    00001E01
             IRLM FUNCTION: LOCK
             LOCK STATE:    S
             LOCK DURATION: INTEREST DURATION FOR P-LOCKS

        REQUEST TYPE/MODE:  UNCONDITIONAL
                            ACQUIRE
                            GROUP UNLOCK
                            RESULTANT STATE
                            NON-MODIFY LOCK
                            PAGE LOCK
                            ASYNCHRONOUS SLM REQUEST
                            GLOBAL LOCK

        REASON FOR SUSPEND: INTER-SYSTEM COMMUNICATION
        REASON FOR RESUME:  NORMAL
  GLOBAL CONTENTION EXTENT: IRLM GLOBAL RESOURCE CONTENTION
```

Figure 114. DTRAC Lock Suspension Pop-Up Display

This pop-up display shows the following:

LOCK TYPE
: Type of locked resource, such as a data page, row, or page set in a table space or index, or an internal lock to control special processing such as utility serialization, data set open, or altering a buffer pool. See Appendix A, "Lock Type Table" on page 405 for a complete list. The long version of the lock type is shown in this pop-up.

RESOURCE #
: Sometimes referred to as the *small resource name*. It consists of a hex number. For example, for a DATA PAGE lock, a physical page number is shown as X'hhhhhh00'. An INDEX PAGE lock on the root page is indicated with the special value of X'000002FF'. A ROW lock is shown as X'hhhhhhrr', where the first six digits are the page number and the last two are the RID. For some resources, this field is blank. See Appendix A, "Lock Type Table" on page 405 for a complete list.

PARENT LOCK TOKEN
: Parent lock token, if one was specified for explicit hierarchical locking.

HASH VALUE
: Hash value of the locked resource name.

IRLM FUNCTION
- LOCK
- UNLOCK
- CHANGE
- QUERY

LOCK STATE

Requested state, which can be one of the following:

UNS	UNPROTECTED SHARE
IS	INTENT SHARE
IX	INTENT EXCLUSIVE
S	SHARED
U	UPDATE
SIX	SHARED INTENT EXCLUSIVE
NSU	NON-SHARED UPDATE
X	EXCLUSIVE

LOCK DURATION
- MANUAL
- TEMPORARY
- COMMIT
- ALLOCATION
- PLAN
- DURATION

REQUEST TYPE/MODE
- CONDITIONAL OR UNCONDITIONAL
- AUTOREL or ACQUIRE
- SINGLE or GROUP UNLOCK
- RESULTANT STATE or COUNT BY STATE
- MODIFY LOCK or NON-MODIFY LOCK
- PAGE LOCK or LOGICAL LOCK
- ASYNCHRONOUS SLM REQUEST or SYNCHRONOUS REQUEST
- GLOBAL LOCK or LOCAL LOCK

REASON FOR SUSPEND
- IRLM LATCH CONTENTION
- IRLM QUEUED REQUEST
- RH TO PTB COMMUNICATION REQUEST
- LOCAL RESOURCE CONTENTION
- GLOBAL RESOURCE CONTENTION
- INTER-SYSTEM COMMUNICATION
- NOTIFY MESSAGE SENT
- LOCAL MEMORY CANNOT BE EXTENDED IN XM MODE

REASON FOR RESUME
- DEADLOCK
- IDENTIFY TO IRLM
- NORMAL
- TIMEOUT

GLOBAL CONTENTION EXTENT
- MVS XES RESOURCE CONTENTION
- IRLM GLOBAL RESOURCE CONTENTION

Lock Timeout/Deadlock Pop-Up (IFCID 054)

Additional data for the LOCK-TMO event is displayed in the Lock Timeout/Deadlock pop-up display, as shown in Figure 115.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC            INPUT   10:18:36  INTVL=> 5  LOG=> N  TGT==> DB2E
PARM ==> POPUPS,SEQ=38,LEVEL=2                        ROW 1 OF 7 SCROLL=> CSR

START: 08:49:40 AUTH: CJN2     PLAN: DSNESPCS CORR: CJN2         CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
    LOCK-TMO     34.651                   *DB=DSNDB06  OB=00000105 TYP=ACE
==============================================================================
             LOCK STATE:    UPDATE

        FIRST HOLDER:
             AUTH ID:       CJN1
             CORR ID:       CJN1
             CONN NAME:     TSO
             PLAN NAME:     DSNESPCS
```

Figure 115. DTRAC Lock Timeout/Deadlock Pop-Up Display

Note: This event does not exist for DB2 version 4 and later.

This pop-up display shows the following:

LOCK STATE
Lock state of the first holder of the lock, which can be one of the following:

INTENT SHARE
INTENT EXCLUSIVE
SHARED
UPDATE
SHARED INTENT EXCLUSIVE
EXCLUSIVE
UNPROTECTED-SHARE
NON-SHARE UPDATE

The following information is available only for contention TYPE=ACE:

FIRST HOLDER
Identification information for the first holder of the lock:

AUTHID	Authorization ID
CORR ID	Correlation ID
CONN NAME	Connection Name
PLAN NAME	Plan name

Page P-Lock Pop-Up (IFCID 259)

Additional data for the PAGE P-LOCK event is displayed in the Page P-Lock pop-up display, as shown in Figure 116.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    14:48:42  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> TEST2BYT,SEQ=2,LEVEL=2                      ROW 1 OF 12 SCROLL=> CSR

START: 12:55:30 AUTH: BOLSMR2  PLAN: DSNTEP41 CORR: BOLSMR4P     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
    PAGE-P-LOCK   3.856                   *DB=DSNDB06  PS=SYSDBASE LOCK
================================================================================

                                  BUFFER POOL:   BP0
                                  OBJECT TYPE:   SPACE MAP PAGE
                              IRLM RETURN CODE:   00000000
                           IRLM RETURN SUBCODE:   00000000
                          REQUESTED LOCK STATE:   X
                                   P-LOCK TYPE:   PAGE P-LOCK
                              PARTITION NUMBER:   0001
                             NEWLY HELD P-LOCK:   X
                        P-LOCK EXIT REASON CODE:   0000
                          P-LOCK REQUEST FLAGS:   UNCONDITIONAL
                                                  NOT RESTART LOCK
                                                  NON-MODIFY LOCK
```

Figure 116. DTRAC Page P-Lock Pop-Up Display

Page P-locks are used for data sharing in DB2 version 4 and later. They are used to preserve inter-DB2 cache page (buffer) coherency when sub-page coherency is used and the page set or partition is shared in R/W mode between two or more DB2 systems. The most common cases of sub-page coherency are as follows:

- Data pages when row level locking is in effect
- Type 2 index leaf pages
- Space map pages

This pop-up display shows the following:

BUFFER POOL
: Buffer pool ID.

OBJECT TYPE
: Type of object locked, which can be one of the following:

  DATA PAGE
  HEADER PAGE
  INDEX PAGE
  SPACE MAP PAGE

IRLM RETURN CODE
: Return code indicating results of the IRLM function.

IRLM RETURN SUBCODE
Return subcode indicating results of the IRLM function.

REQUESTED LOCK STATE
P-lock state requested for lock or change request, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

P-LOCK TYPE
Type of P-lock, which can be as follows:

PAGE P-LOCK

PARTITION NUMBER
Partition number if this is a partitioned page set. (This field is zero if this is a nonpartitioned page set.)

NEWLY HELD P-LOCK
Newly held p-lock state, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

P-LOCK EXIT REASON CODE
Reason code, which indicates the reason for P-lock exit.

P-LOCK REQUEST FLAGS
Three p-lock request flags are shown. They can be as follows:

CONDITIONAL or UNCONDITIONAL

RESTART LOCK or NOT RESTART LOCK

MODIFY LOCK or NON-MODIFY LOCK

Package Allocation Pop-Up (IFCID 177)

Additional data for the PKG-ALLOC event is displayed in the Package Allocation pop-up display, as shown in Figure 117.

```
BMC SOFTWARE --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    10:05:12  INTVL=> 5  LOG=> N  TGT==> DB2KSR
PARM ==> DET4,SEQ=36,LEVEL=2                          ROW 1 OF 14 SCROLL=> CSR

START: 09:30:17 AUTH: BOLCJN3  PLAN: DSNESPRR CORR: BOLCJN3      CONN: TSO
===============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
PKG-ALLOC         0.356                   *DSNESPRR ISO=RR ACQ=USE   REL=COMIT
===============================================================================
        LOCATION:      DB2K
        COLLECTION-ID: DSNESPRR
        PACKAGE-ID:    DSNESM68
        DYNAMIC RULES: RUN
        VERSION:

        DEFER:         NO
        REOPTIMIZE:    NO
        KEEP DYN STMTS:NO
        DB PROTOCOL:   DRDA
        OPTIMIZE HINT:
      IMMEDIATE WRITE: NO
      ENCODING SCHEME: EBCDIC
       ENCODING CCSID: 37
```

Figure 117. DTRAC Package Allocation Pop-Up Display

This pop-up display shows the following:

LOCATION
: Location of package. This is blank if the local location is unnamed.

COLLECTION-ID
: Collection ID of the package.

PACKAGE-ID
: Package ID of the package.

DYNAMIC RULES
: DYNAMICRULES BIND option for dynamic SQL statements. It can be:

  BIND — Bind-time rules apply for dynamic SQL statements at run time.

  RUN — Run-time rules apply for dynamic SQL statements at run time.

  NONE — DYNAMICRULES was not specified. The package has the DYNAMICRULES option value of the plan to which it is appended at run time.

VERSION
: Version identification name.

DEFER
: YES|NO. YES indicates preparation of dynamic SQL statements is deferred. NO indicates they are prepared immediately.

REOPTIMIZE
YES|NO. NO indicates access paths are chosen only at bind or rebind time. YES indicates they are determined again at execution time.

Access paths can be determined at execution time for statements with the following types of variable input:

- Input host variables
- Parameter markers
- Values in special registers

KEEP DYN STMTS
YES|NO. Indicates whether DB2 keeps (YES) or discards (NO) prepared SQL statements at commit points.

DB PROTOCOL
Database protocol for three-part names, which can be DRDA or PRIVATE.

OPTIMIZE HINT
Optimization hints value, or blanks if no optimization hints are defined.

IMMEDIATE WRITE
YES|NO|PH1. Specifies immediate write of updated group buffer pool buffers.

ENCODING SCHEME
Encoding scheme of the encoding CCSID, which can be one of the following:

ASCII
EBCDIC
UNICODE

ENCODING CCSID
Encoding CCSID value. DB2 derives this value from the value of the encoding bind option.

Parallel Degree Pop-Up (IFCID 221)

Additional data for the PRLL-DEGREE event is displayed in the Parallel Degree pop-up display, as shown in Figure 118.

Note: Not all fields in the following sample are shown each time this pop-up display is accessed.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    11:10:42  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> PRLLPLEX,SEQ=110,LEVEL=3                    ROW 1 OF 15 SCROLL=> CSR

START: 06:18:22 AUTH: BOLSMR4  PLAN: SMRTEP5A CORR: BOLSMR4B     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 PRLL-DEGREE     31.987                   *PLANNED(RUN)=109 ACTUAL=109 GROUP=1
==============================================================================
             LOCATION:       DB2H
             PKG COLL ID:    SMRTEP2A
             PROGRAM:        SMRTEP2
             STATEMENT:      1,224
             QUERY BLOCK:    1
             PLANNED(BIND):  109
             DEGREE REASON:  PLANNED BIND TIME DEGREE
     RESOURCE EXPLOITATION:  CPU
 MEMBERS QUERY EXECUTE ON:   1
           NUMBER SECTIONS:  1

  PARTITION STATUS:             NORMAL
      LOW PAGE:           1
      HIGH PAGE:          1,049K
      LOW KEY:    00000000  00000000  00000000  00000000   *................*
      HIGH KEY:   00000000  00000000  00000000  00000000   *................*

   DB2 MEMBER NAME: DB2     BUFFER POOL CONSTRAINED: NO

   ** BUFFER POOL CONSTRAINT INFORMATION **
      BPID: BP11       TABLESPACE
      DBID: 00000107    SMR8D51A
      PSID: 00000002    SMR8BP11
```

Figure 118. DTRAC Parallel Degree Pop-Up Display

This pop-up display shows the following:

LOCATION
: Location name.

PKG COLL ID
: Package collection-ID.

PROGRAM
: Program name.

STATEMENT
: The statement number.

QUERY BLOCK
: Query block number.

PLANNED(BIND)
: Planned degree of parallel I/O processing at BIND time.

Parallel I/O processing decisions are made at BIND time. However, this field contains zeros if the statement has host variables. Host variables cause the parallel I/O processing decision to be made at run time.

DEGREE REASON
Reason for deriving the planned degree of parallel I/O processing at run time, which can be one of the following:

NORMAL CONDITION
HOST VARIABLE PARTITIONING
NO ESA SORT SUPPORT
THE CURSOR MIGHT BE USED FOR UPDATE OR DELETE
THE PARALLEL GROUP IS EMPTY

RESOURCE EXPLOITATION
Type of resource exploitation, which can be one of the following:

CPU
CPU exploitation

I/O
I/O bandwidth exploitation

blank
None

MEMBERS QUERY EXECUTE ON
Number of members on which a query executed.

NUMBER SECTIONS
Number of repeating data sections in this trace record.

PARTITION STATUS
Status of this partition range, which can be one of the following:

NORMAL condition
A parallel task is created for this partition range.

EMPTY partition range
No parallel task is created for this partition range.

LOW PAGE
Low page number of the page range for the partition, if the partitioning scheme is using a page range. Use of page partitioning is determined by DB2.

HIGH PAGE
High page number of the page range for the partition, if the partitioning scheme is using a page range. Use of page partitioning is determined by DB2.

LOW KEY
First 240 bytes of the low boundary key range, if the partitioning scheme is using a key range. (This is the key range on the table definition).

HIGHKKEY
First 240 bytes of the high boundary key range, if the partitioning scheme is using a key range. (This is the key range on the table definition.)

DB2 MEMBER NAME
Each DB2 member name and whether or not the buffer pool is constrained.

Buffer Pool Constraint Information:

BPID
Buffer pool ID and the type of page set.

DBID
Database ID of an object involved in a parallel access or join.

PSID
Page set ID of an object involved in a parallel access or join.

Parallel Elapsed Pop-Up (IFCID 222)

Additional data for the PRLL-ELAP event is displayed in the Parallel Elapsed pop-up display, as shown in Figure 119.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    11:12:24  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> PRLLPLEX,SEQ=110,LEVEL=3                    ROW 1 OF 79 SCROLL=> CSR

START: 06:18:22 AUTH: BOLSMR4  PLAN: SMRTEP5A CORR: BOLSMR4B     CONN: BATCH
===============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 PRLL-ELAP      306.218                   *ELAP=281.382  ROWS=109
===============================================================================
             LOCATION:       DB2H
             PKG COLL ID:    SMRTEP2A
             PROGRAM:        SMRTEP2
             STATEMENT:      1,224
             QUERY BLOCK:    1
             PARALLEL GROUP: 1
             PIPE START:     1997-11-18-06.18.41.647931
             PIPE END:       1997-11-18-06.23.23.029507
             ELAPSED:        281.382


                   ** SUBPIPE  DATA SECTION **
 CONSUMED CHILD TASK PRODUCED  DB2
 ROWS      SUSPENDS  ROWS      MEMBER   START         END           ELAPSED
 1         0         1         DB2H     06.18.41.666  06.20.50.598  128.932
 1         0         1         DB2H     06.18.44.247  06.20.55.781  131.534
 1         0         1         DB2H     06.18.41.690  06.22.12.947  211.257
 1         0         1         DB2H     06.18.42.709  06.20.53.515  130.806
 1         0         1         DB2H     06.18.44.079  06.20.56.739  132.660
 1         0         1         DB2H     06.18.44.317  06.22.48.663  244.347
 1         0         1         DB2H     06.18.44.435  06.21.08.069  143.634
 1         0         1         DB2H     06.18.44.315  06.20.58.558  134.243
```

Figure 119. DTRAC Parallel Elapsed Pop-Up Display

This pop-up display shows the following:

LOCATION
: Location name.

PKG COLL ID
: Package collection-ID.

PROGRAM
: Program name (package or DBRM name).

STATEMENT
: Statement number.

QUERY BLOCK
: Query block number.

PARALLEL GROUP
: Parallel group number.

PIPE START
Timestamp of pipe creation (in DB2 timestamp format). A pipe is the data structure that is created when the main task sets up for parallel I/O processing.

PIPE END
Timestamp of pipe termination (in DB2 timestamp format). This is when all parallel I/O processing for this task ends.

ELAPSED
Time difference between PIPE START and PIPE END.

Subpipe Data Section:

A subpipe is a data structure used for one of the individual groups in an operation using parallel I/O processing.

CONSUMED ROWS
Number of rows consumed for this subpipe.

CHILD TASKS SUSPENDED
Number of times the child task was suspended while waiting for the parent task.

PROCESSED|PRODUCED|INPUT ROWS
Number of rows in the input work file or the total number of rows processed or produced for this subpipe.

DB2 MEMBER
Name of the DB2 member that supplies the data (DB2 5.1 and later).

START
Timestamp of subpipe creation (in DB2 timestamp format).

END
Timestamp of subpipe termination (in DB2 timestamp format).

ELAPSED
Time difference between SUBPIPE START and SUBPIPE END.

Parallel Group Pop-Up (IFCID 231)

Additional data for the PRLL-GROUP event is displayed in the Parallel Group pop-up display, as shown in Figure 120.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    06:20:18  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> PRLL231,SEQ=11,LEVEL=3                      ROW 1 OF 19 SCROLL=> CSR

START: 13:50:14 AUTH: BOLSMR4  PLAN: SMRTEP5A CORR: BOLSMR4B     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
  PRLL-GROUP     19.666                   *ELAP=19.649   GROUP=1
==============================================================================
            STATEMENT NUMBER:  1,224
          QUERY BLOCK NUMBER:  1
              GROUP CREATION:  1997-12-02-13.50.15.061715
           GROUP TERMINATION:  1997-12-02-13.50.34.711102

                    *** TASK  DATA SECTION ***
 TASK     DB2      CREATE          TERMINATE       SERVICE     CPU
 SEQUENCE MEMBER    TIME            TIME            UNITS     TIME
 1        DB2H     13.50.15.177    13.50.33.859    73          76 ms
 2        DB2H     13.50.15.197    13.50.34.631    75          79 ms
 3        DB2H     13.50.15.203    13.50.34.195    73          77 ms
 4        DB2H     13.50.15.213    13.50.34.547    77          81 ms
 5        DB2H     13.50.15.222    13.50.34.377    64          67 ms
 6        DB2H     13.50.15.231    13.50.34.646    63          66 ms
 7        DB2H     13.50.15.239    13.50.34.349    74          78 ms
 8        DB2H     13.50.15.249    13.50.34.686    78          82 ms
 9        DB2H     13.50.15.257    13.50.33.728    71          75 ms
 10       DB2H     13.50.15.278    13.50.33.592    71          75 ms
 11       DB1H     13.50.15.336    13.50.34.704    15          17 ms
```

Figure 120. DTRAC Parallel Group Pop-Up Display

This pop-up display shows the following:

STATEMENT NUMBER
: Number of the SQL statement.

QUERY BLOCK NUMBER
: Number of the query block.

GROUP CREATION
: Timestamp of the creation of the parallel group.

GROUP TERMINATION
: Timestamp of the termination of the parallel group.

Task Data Section:

TASK SEQUENCE
Task sequence number.

DB2 MEMBER
Name of the DB2 member on which the task executes.

CREATE TIME
Timestamp of task creation.

TERMINATE TIME
Timestamp of task termination.

SERVICE UNITS
Task CPU service units consumed.

CPU TIME
Task CPU execution time.

Plan Allocation Pop-Up (IFCID 112)

Additional data for the PLAN-ALLOC event is displayed in the Plan Allocation pop-up display, as shown in Figure 121.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==>DTRAC INPUT  10:02:21 INTVL=>5LOG=>NTGT==>DB2J
PARM ==>POPUP2,SEQ=41,LEVEL=2                        ROW 1 OF 6SCROLL=>CSR

START: 08:00:17 AUTH: BOLCJN2  PLAN: DSNBIND  CORR: CJNBCJN      CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 PLAN-ALLOC       0.000                   *ISOLATION=CS ACQ=ALLOC REL=DEALLOC
==============================================================================
        DYNAMIC RULES: RUN
        DEFER:         NO
        REOPTIMIZE:    NO
        KEEP DYN STMTS:NO
        DB PROTOCOL:   PRIVATE
        OPTIMIZE HINT:
```

Figure 121. DTRAC Plan Allocation Pop-Up Display

This pop-up display shows the following:

DYNAMIC RULES
DYNAMICRULES BIND option for dynamic SQL statements. It can be:

BIND Bind-time rules apply for dynamic SQL statements at run time.

RUN Run-time rules apply for dynamic SQL statements at run time.

NONE DYNAMICRULES was not specified. The package has the DYNAMICRULES option value of the plan to which it is appended at run time.

DEFER
YES|NO. YES indicates that preparation of dynamic SQL statements is deferred. NO indicates that they are prepared immediately.

REOPTIMIZE
YES|NO. NO indicates that access paths are chosen only at bind or rebind time. YES indicates that they are determined again at execution time.

Access paths can be determined at execution time for statements with the following types of variable input:

- Input host variables
- Parameter markers
- Values in special registers

KEEP DYN STMTS
YES|NO. Indicates whether DB2 keeps (YES) or discards (NO) prepared SQL statements at commit points.

DB PROTOCOL
Database protocol for three-part names, which can be DRDA or PRIVATE.

OPTIMIZE HINT
Optimization hints value, or blanks if no optimization hints are defined.

P-Lock Pop-Up (IFCID 251)

Additional data for the P-LOCK event is displayed in the P-Lock pop-up display, as shown in Figure 122.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    13:04:36  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> TEST2BYT,SEQ=1,LEVEL=2                      ROW 1 OF 15 SCROLL=> CSR

START: 12:55:01 AUTH: BOLSMR2  PLAN: DSNTEP41 CORR: BOLSMR4D     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
  P-LOCK          2.158                   *DB=DSN8D41A PS=DSN8S41E LOCK
================================================================================

                                  BUFFER POOL:   BP0
                                  OBJECT TYPE:   TABLESPACE
                             IRLM RETURN CODE:   00000004
                          IRLM RETURN SUBCODE:   10000004
                         REQUESTED LOCK STATE:   X
                                  P-LOCK TYPE:   PARTITION
                             PARTITION NUMBER:   0001
                       PREVIOUSLY HELD P-LOCK:   S
                       PREVIOUS P-LOCK CACHED:   S
                            NEWLY HELD P-LOCK:   X
                      NEW P-LOCK CACHED STATE:   X
                           P-LOCK REASON CODE:   0000
                         P-LOCK REQUEST FLAGS:   UNCONDITIONAL
                                                 NOT RESTART LOCK
                                                 NON-MODIFY LOCK
```

Figure 122. DTRAC P-Lock Pop-Up Display

Physical locks or P-locks are used for data sharing in DB2 version 4 and later. They are used for a non-partitioned table space, a non-partitioned index, a partition of a partitioned table space, and a partition of a partitioned index. DB2 Buffer Manager uses P-locks on these objects to track the inter-DB2 interest levels. The P-lock cache state determines the inter-DB2 interest level on the object.

This pop-up display shows the following:

BUFFER POOL
: Buffer pool ID.

OBJECT TYPE
: Type of object, which can be one of the following:

  TABLESPACE
  INDEX

IRLM RETURN CODE
: Return code indicating results of the IRLM function.

IRLM RETURN SUBCODE
: Return subcode indicating results of the IRLM function.

REQUESTED LOCK STATE
P-lock state requested for lock or change request, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

P-LOCK TYPE
Type of P-lock, which can be one of the following:

PARTITION
PAGESET

PARTITION NUMBER
Partition number if this is a partitioned page set. (This field is zero if this is a nonpartitioned page set.)

PREVIOUSLY HELD P-LOCK
Previously held P-lock state, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

PREVIOUS P-LOCK CACHED
Previous P-lock cached state, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

NEWLY HELD P-LOCK
Newly held p-lock state, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

NEW P-LOCK CACHED STATE
New P-lock cached state, which can be one of the following:

IS This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

IX This DB2 has R/W interest in the object and one or more other DB2 members have R/W interest.

S This DB2 has R/O interest in the object and one or more other DB2 members have R/W interest.

SIX This DB2 has R/W interest in the object and one or more other DB2 members have R/O interest.

X This DB2 has R/W interest in the object and no other DB2 member has declared interest.

P-LOCK REASON CODE
P-lock exit reason code.

P-LOCK REQUEST FLAGS

Three p-lock request flags are shown. They can be as follows:

CONDITIONAL or UNCONDITIONAL

RESTART LOCK or NOT RESTART LOCK

MODIFY LOCK or NON-MODIFY LOCK

Read I/O Pop-Up (IFCID 006/007)

Additional data for the READ-I/O event is displayed in the Read I/O pop-up display, as shown in Figure 123.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    12:35:34  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TRACE1,SEQ=000002,LEVEL=3            ROW     1 OF     2 SCROLL=> CSR

START: 12:26:16 AUTH: BABUSER  PLAN: DSNTIA21 CORR: DB221REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
    READ-I/O     10.052    62 ms   404 us *DB=DEMODB32 TS=T32      BP32 SYNC-RD
==============================================================================
             RETURN CODE:    0
             PAGES READ:     1
```

Figure 123. DTRAC Read I/O Pop-Up Display

This pop-up appears for any of the following read requests:

- Synchronous read of one page
- Sequential prefetch
- List prefetch
- Dynamic sequential detection

It shows the following:

RETURN CODE:
Return code from media manager

PAGES READ:
Number of pages read

Requesting Agent Data Pop-Up (IFCID 183)

Additional data for the REQ-AGNT event is displayed in the Requesting Agent Data pop-up display, as shown in Figure 124.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    19:01:59  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=1,LEVEL=2                              ROW 1 OF 13 SCROLL=> CSR

START: 14:14:21 AUTH: CMRL     PLAN: LILYREP  CORR: CMRLJOB      CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
REQ-AGNT         15.409 1,698 ms   197 ms *RC(0)      FETCH
==============================================================================
PROGRAM:         DSNESM68
LOCATION:        DB1D
COLLECTION-NAME: BBXCOL3
SQL REQUEST:     FETCH
STMT NUMBER:     137
RETURN CODE:     00000000
```

Figure 124. DTRAC Requesting Agent Data Pop-Up Display

This pop-up display shows the following:

PROGRAM
Program name.

LOCATION
Application server location name.

COLLECTION-NAME
Collection ID.

SQL REQUEST
SQL statement type. Possible values include the following:

OPEN
FETCH
CLOSE
PREPARE
EXECUTE
EXECUTE IMMEDIATE
DESCRIBE
EXPLAIN
SET CURRENT SQLID
SELECT
INSERT
DELETE
UPDATE
SELECT
GRANT
REVOKE
ROLLBACK
LOCK
CREATE VIEW
COMMIT

CREATE DATABASE
CREATE TABLESPACE
CREATE STOGROUP
CREATE TABLE
CREATE INDEX
CREATE SYNONYM
DROP VIEW
DROP SYNONYM
DROP INDEX
DROP TABLE
DROP TABLESPACE
DROP DATABASE
DROP STOGROUP
ALTER STOGROUP
ALTER TABLESPACE
ALTER INDEX
ALTER TABLE
COMMENT ON
LABEL ON
DROP ALIAS
DROP PACKAGE
ALTER DATABASE
CONNECT TO
CONNECT RESET
CONNECT
CONNECT TO (TYPE 2)
CONNECT RESET (TYPE 2)
CONNECT (TYPE 2)
SET CONNECTION
RELEASE
RELEASE CURRENT
RELEASE ALL
RELEASE ALL SQL
RELEASE ALL PRIVATE

STMT NUMBER
Statement number of the SQL statement.

RETURN CODE
Return code from the SQL statement.

Requesting Site—DRDS Pop-Up (IFCID 157)

Additional data for the REQ-DRDS event is displayed in the Requesting Site—DRDS pop-up display, as shown in Figure 125.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:18:40  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=2                                ROW 1 OF 8 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 REQ-DRDS         7.959 2,968 ms 1,121 us *TYPE=SETUPCAL LOC=DB1D
================================================================================

DRDS REQUEST AT REQUESTING SITE

                  REQUEST TYPE: WRITE

                  PLAN SECTION: 1

                  PROGRAM:      DSNESM68
```

Figure 125. DTRAC Requesting Site—DRDS Pop-Up Display

This pop-up display shows the following:

REQUEST TYPE
Type of request:

READ
WRITE

PLAN SECTION
Section number in plan

PROGRAM
Program name

Requesting Site—DTM (IFCID 162)

Additional data for the REQ-DTM event is displayed in the Requesting Site—DTM pop-up display, as shown in Figure 126.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:18:13  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=2                                ROW 1 OF 6 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 REQ-DTM          7.958                   *TYPE=MIGRATE TRANSACTION
==============================================================================

DISTRIBUTED TRANSACTION MANAGER REQUEST AT REQUESTING SITE

                  LOCATION:      DB1D

                  LOCATION TYPE: READ LOCATION
```

Figure 126. DTRAC Requesting Site—DTM Pop-Up Display

This pop-up display shows the following:

LOCATION
Sixteen-byte location name

LOCATION TYPE
Type of locations:

READ LOCATION
UPDATE LOCATION
VOTE ‘NO’ RESPONSE
ACKNOWLEDGE COMMIT
READ RESPONSE
ERROR

This field is displayed only if TYPE=COMMIT MESSAGE SENT or TYPE=RESPONSE RECEIVED.

Requesting Site—VTAM Pop-Up (IFCID 160)

Additional data for the REQ-VTAM event is displayed in the Requesting Site—VTAM pop-up display, as shown in Figure 127.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:21:24  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=3                              ROW 1 OF 18 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 REQ-VTAM         3.817                   *TYPE=CREATE CONVERSATION
================================================================================

VTAM REQUEST AT REQUESTING SITE

                  MESSAGE TYPE:
                  MESSAGE RESPONSE:

                  MESSAGE LENGTH:   0
                  MESSAGE CLASS:    0
                  MESSAGE NUMBER:   0

                  INSTANCE ID:      X'040C000081140B38'
                  CONVERSATION ID:  X'00000000'
                  VTAM CONV ID:     X'00FD35F0'
                  LOG MODE NAME:    ........

             MESSAGE TIME STAMP:
             ELAPSED TIME IN VTAM:    144 us
             CPU TIME OF DB AGENT:      0 us
```

Figure 127. DTRAC Requesting Site—VTAM Pop-Up Display

This pop-up display shows the following:

MESSAGE TYPE
Type can be one of the following:

REQUEST MESSAGE
BROADCAST MESSAGE
DATAGRAM MESSAGE
IDENTIFY MESSAGE

MESSAGE RESPONSE
Response can be one of the following:

DATA MESSAGE
CONFIRMATION MESSAGE
NO MESSAGE

MESSAGE LENGTH
Length of message in bytes

MESSAGE CLASS
Class of message

MESSAGE NUMBER
Number of message

INSTANCE ID
Eight-byte VTAM session instance identifier in hexadecimal

CONVERSATION ID
Four-byte logical connection identifier in hexadecimal

VTAM CONV ID
Four-byte VTAM APPC conversation identifier in hexadecimal

LOG MODE NAME
Log mode entry name from communications database (not always available)

MESSAGE TIME STAMP
Timestamp when message was sent or received

ELAPSED TIME IN VTAM
Elapsed time in VTAM

CPU TIME OF DB AGENT
CPU time of database access agent sent from serving site

RID List Pop-Up (IFCID 125)

Additional data for the RIDLIST event is displayed in the RID List pop-up display, as shown in Figure 128.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==>DTRAC INPUT  13:16:45 INTVL=>5LOG=>NTGT==>DB2J
PARM ==>DTL0229,SEQ=9,LEVEL=2                       ROW 1 OF 8SCROLL=>CSR
                                                          HIST TGT--- DB2Y
START: 23:42:56 AUTH: BOLCJN2  PLAN: RXDB2    CORR: BOLCJN2      CONN: DB2CALL
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 RIDLIST          0.288                   *RIDS IN FINAL LIST(1)
================================================================================
                 PACKAGE COLLECTION ID:  RXD2
                               PLAN ID:  RXDB2
                          PROGRAM NAME:  RXSEL1M
                  PRECOMPILE TIMESTAMP:  1578F8620304C9F4
                  STATEMENT NUMBER:        226

               DATABASE     INDEX  THRESHOLD     #RIDS
               DSNDB06   00000066      3,260         1
```

Figure 128. DTRAC RID List Pop-Up Display

This pop-up display shows the following:

PACKAGE COLLECTION ID
: Package collection ID for this query.

PLAN ID
: Plan ID for this query.

PROGRAM NAME
: Program name for this query.

PRECOMPILE TIMESTAMP
: Precompiler timestamp for this query.

STATEMENT NUMBER
: Statement number for this query.

DATABASE
: Name of the database containing the referenced indexes.

INDEX
: OBID (Object ID) of this index. (If this trace is DETAIL2 or higher, you can refer to the preceding scan events for the name of the table space corresponding to this index OBID.)

THRESHOLD
: Threshold value for this index of maximum number of RIDs to process with list prefetch before switching to a sequential scan.

#RIDS

Number of RIDs processed in this step. If the MIAP (multiple index access path) processed failed during this step, one of the following messages appears instead:

RETRIEVAL SKIPPED
NO RID STORAGE
RIDS > MAXIMUM LIMIT

Scan Pop-Up (IFCID 017/018)

Additional data for the INDX-SCAN, FRST-INSRT, and SEQ-SCAN events is displayed in the Scan pop-up display, as shown in Figure 129.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==>DTRAC INPUT  11:07:59 INTVL=>5LOG=>NTGT==>DB2J
PARM ==>ZAP1858,SEQ=1,LEVEL=3                       ROW 1 OF 15SCROLL=>CSR

START: 14:37:21 AUTH: BOLCJN2  PLAN: RXDB2    CORR: BOLCJN2      CONN: DB2CALL
===============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 INDX-SCAN        0.203    31 ms   542 us *TS=SYSDBASE TB=SYSTABLES
===============================================================================
                                      INDEX SEQ-DATA
                                   -------- --------
ROWS PROCESSED ALL TYPES                  1        1
ROWS PROCESSED CORRECT TYPE               0        1
ROWS QUALIFIED BY DM (STAGE 1)            1        0
ROWS QUALIFIED BY RDS (STAGE 2)           0        0
ROWS INSERTED                             0        0
ROWS UPDATED                              0        0
ROWS DELETED                              0        0
PAGES SCANNED                             2        1
REFERENTIAL INTEGRITY PROCESSING:
 PAGES SCANNED                            0        0
 ROWS DELETED/SET NULL                    0        0
LOB- PAGES SCANNED                        0        0
   - PAGES UPDATED                        0        0
```

Figure 129. DTRAC Scan Pop-Up Display

A maximum of five columns can be displayed showing the individual statistics for index (INDEX), sequential data (SEQ-DATA), work tables (SEQ-WORK), temporary tables (SEQ-TEMP), and transition tables (SEQ-TRAN). Only columns for the types of scan used are displayed.

ROWS PROCESSED ALL TYPES
: Number of rows processed in the table space of all types. This figure is meaningful when a table space contains more than one table. Rows for other tables must also be accessed to determine whether they are required to satisfy the request.

ROWS PROCESSED CORRECT TYPE
: Number of rows processed that were of the correct type. For a single table space, this should equal ROWS PROCESSED ALL TYPES.

ROWS QUALIFIED BY DATA MANAGER
: Number of rows that qualified for the request by DB2's Data Manager component.

ROWS QUALIFIED BY RDS
: Number of rows that qualified for the request by DB2's Relational Database Support component.

ROWS INSERTED
: Number of rows inserted.

ROWS UPDATED
Number of rows updated.

ROWS DELETED
Number of rows deleted.

PAGES SCANNED
Number of database pages that needed to be scanned to satisfy the request.

REFERENTIAL INTEGRITY PROCESSING:
Data displayed includes the following:

PAGES SCANNED
Number of pages scanned to enforce referential integrity constraints.

ROWS DELETED/SET NULL
Number of rows deleted or set to a null value to enforce referential integrity.

LOB- PAGES SCANNED
Number of additional pages scanned in a LOB table space.

LOB- PAGES UPDATED
Number of LOB data pages updated by an SQL INSERT or UPDATE.

Serving Site—DRDS Pop-Up (IFCID 158)

Additional data for the SERV-DRDS event is displayed in the Serving Site—DRDS pop-up display, as shown in Figure 130.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:27:39  INTVL=> 3  LOG=> N  TGT==> DB1D
PARM ==> ,SEQ=1,LEVEL=3                                ROW 1 OF 6 SCROLL=> CSR

START: 09:11:51 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
SERV-DRDS         0.004              0 us *TYPE=SETUPCAL
================================================================================

DRDS REQUEST FOR DATA AT SERVING SITE

                  PLAN SECTION: 1

                  PROGRAM:      DSNESM68
```

Figure 130. DTRAC Serving Site—DRDS Pop-Up Display

This pop-up display shows the following:

PLAN SECTION
: Section number in plan

PROGRAM
: Program name

Serving Site—DTM Pop-Up (IFCID 163)

Additional data for the SERV-DTM event is displayed in the Serving Site—DTM pop-up display, as shown in Figure 131.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:55:24  INTVL=> 3  LOG=> N  TGT==> DB1D
PARM ==> ,SEQ=6,LEVEL=3                              ROW 1 OF 13 SCROLL=> CSR

START: 10:52:41 AUTH: CIR2     PLAN: DSNESPCS CORR: CIR2         CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
SERV-DTM          0.991                   *TYPE=MIGRATE TRANSACTION RECEIVED
================================================================================

DISTRIBUTED TRANSACTION MANAGER REQUEST AT SERVING SITE
```

Figure 131. DTRAC Serving Site—DTM Pop-Up Display

This pop-up display shows an expansion of the DTM request at the serving site.

Serving Site—VTAM Pop-Up (IFCID 161)

Additional data for the SERV-VTAM event is displayed in the Serving Site—VTAM pop-up display, as shown in Figure 132.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    10:55:24  INTVL=> 3  LOG=> N  TGT==> DB1D
PARM ==> ,SEQ=6,LEVEL=3                              ROW 1 OF 13 SCROLL=> CSR

START: 10:52:41 AUTH: CIR2     PLAN: DSNESPCS CORR: CIR2         CONN: TSO
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
SERV-VTAM         0.991                   *TYPE=RECEIVE CONVERSATION
==============================================================================

VTAM REQUEST AT SERVING SITE


               MESSAGE TYPE:     REQUEST MESSAGE

               MESSAGE RESPONSE: DATA MESSAGE
               MESSAGE LENGTH:   0
               MESSAGE CLASS:    0
               MESSAGE NUMBER:   0

               INSTANCE ID:      X'040C000081140B38'
               CONVERSATION ID:  X'00000000'
               VTAM CONV ID:     X'00FD35F0'
               LOG MODE NAME:    ........

           MESSAGE TIME STAMP:

           ELAPSED TIME IN VTAM:      144 us

           CPU TIME OF DB AGENT:        0 us
```

Figure 132. DTRAC Serving Site—VTAM Pop-Up Display

This pop-up display shows the following:

MESSAGE TYPE
Type can be one of the following:

REQUEST MESSAGE
BROADCAST MESSAGE
DATAGRAM MESSAGE
IDENTIFY MESSAGE

MESSAGE RESPONSE
Type can be one of the following:

DATA MESSAGE
CONFIRMATION MESSAGE
NO MESSAGE

MESSAGE LENGTH
Length of message in bytes

MESSAGE CLASS
Class of message

MESSAGE NUMBER
: Number of message

INSTANCE ID
: Eight-byte VTAM session instance identifier in hexadecimal

CONVERSATION ID
: Four-byte logical connection identifier in hexadecimal

VTAM CONV ID
: Four-byte VTAM APPC conversation identifier in hexadecimal

LOG MODE NAME
: Log mode entry name from communications database (not always available)

MESSAGE TIME STAMP
: Timestamp when message was sent or received

ELAPSED TIME IN VTAM
: Elapsed time in VTAM

CPU TIME OF DB AGENT
: CPU time of database access agent sent from serving site

Sort Pop-Up (IFCID 95/96)

Additional data for the SORT event is displayed in the Sort pop-up display, as shown in Figure 133.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC            INPUT   10:02:23  INTVL=> 5  LOG=> N  TGT==> DB2F
PARM ==> TRAUTH,SEQ=1,LEVEL=2                        ROW 1 OF 14 SCROLL=> CSR

START: 17:06:41 AUTH: T127     PLAN: DSN8CC0  CORR: GT00D8CS     CONN: CICS332B
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 SORT            14.807   172 MS    31 MS *REC(     1) RECSZ(   74) WKFILE(  1)
==============================================================================
             RETURN CODE:                0
             INITIAL RUNS:               1
             ROWS DELETED:               0
             MERGE PASSES:               0
             SORT TYPE:                  ESA SORT HARDWARE
             PROGRAM NAME:               DSN8CC2
             PACKAGE ID:                 DSN8CC31
             STATEMENT NUMBER:           1,465
             PARTITIONED WORKFILES:      0
             PARTITIONING:               NO
             OUTPUT SORTED/PARTITIONED:  NO
             PARTITIONED:                UNKNOWN
             SORT COLUMNS:               6
             SORT KEYS:                  1
```

Figure 133. DTRAC Sort Pop-Up Display

This pop-up display shows the following:

RETURN CODE
: Sort return code.

INITIAL RUNS
: Number of initial runs.

The sorting records can take more than one run. The number of runs needed depends on the distribution of sort key values. The number of runs is equal to the number of work files created in the input phase (does not include the output work file when there is more than one work file) for MVS/ESA 3.1.3. The maximum number of runs is limited only by the buffer pool size.

ROWS DELETED
: Number of rows deleted because records were merged for the evaluation of column functions with “GROUP BY.”

MERGE PASSES
: Number of MERGE PASSES during sort processing.

SORT TYPE
Type of SORT that occurred, which can be one of the following:

- Using ESA SORT facility hardware instructions
- Group by
- Regular (not ESA, group by, or tag)
- Tag
- ESA tag
- No SORT occurred

PROGRAM NAME
Program name for the query that invokes SORT.

PACKAGE ID
Package collection-ID for the query that invokes SORT.

STATEMENT NUMBER
Statement number for the query that invokes SORT.

PARTITIONED WORKFILES
Number of work files (equal to the degree of parallel I/O processing) that SORT partitioned.

PARTITIONING
Indicates partitioning of the sorted records, as follows:

Y The sorted records are being partitioned.
N The sorted records are not being partitioned.

OUTPUT SORTED/PARTITIONED
Indicates whether an output work file is sorted as well as partitioned, as follows:

Y It is sorted.
N It is not sorted.

PARTITIONED
Indicates when the partitioning took place, as follows:

- Partitioning occurred at the end of the input phase and no merge occurred (WORKFILE).
- Partitioning occurred during the last merge pass.
- Only one record was put into one partition.
- The records were presorted before being partitioned.

SORT COLUMNS
Number of sort columns.

SORT KEYS
Number of sort keys.

SQL Statement Pop-Up

Additional data for SQL statement events is displayed in the SQL Statement pop-up display, as shown in Figure 134.

```
BMC Software --------------   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
       LOB- PAGES SCANNED                          0         0
          - PAGES UPDATED                          0         0
```

Figure 134. DTRAC SQL Statement Pop-Up Display

Expand: From the DTRAC SQL Statement pop-up display you can expand to:

SQLTEXT(EXPLAIN)

A display of the complete statement text, if the statement is static SQL and can be explained and if RxD2 is installed.

From this panel you can choose to:

- EXPLAIN the text and access the RxD2 EXPLAIN display for the statement

  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.

- 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.

- 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: Access is possible to remote DB2 subsystems only if connected with DDF to a local DB2 (in the same MVS system as the TS).

Description: A maximum of five columns can be displayed showing the individual statistics for index (INDEX), sequential data (SEQ-DATA), work tables (SEQ-WORK), temporary tables (SEQ-TEMP), and transition tables (SEQ-TRAN). Only columns for the types of scan used are displayed.

Each row describes the type of processing that DB2 performed to satisfy the SQL statement by:

CURSOR:
Name of the cursor used in the application program. This field can be blank. If it is not blank, it is displayed as shown in Figure 134 on page 356.

LOCATION:
Name of the location associated with the thread. This field can be blank. If it is not blank, it is displayed as shown in Figure 134 on page 356.

ISOLATION:
Isolation level of the statement, which can be

REPEATABLE READ
CURSOR STABILITY
READ STABILITY
UNCOMMITTED READ

This field can be blank. If it is not blank, it is displayed as shown in Figure 134 on page 356.

REOPTIMIZ:
Indicates that DB2 reoptimized the access path. This field is blank if DB2 did not reoptimize. If it is not blank, it is displayed as shown in Figure 134 on page 356.

PACKAGE:
Package collection ID. This field can be blank. If it is not blank, it is displayed as shown in Figure 134 on page 356. The package name is in a special compressed format. This compressed value is carried in the detail trace record and is decompressed only when it is displayed.

PROGRAM:
Name of the program that issued the SQL statement. If this field contains a series of periods (.), the program name detected is unidentifiable.

ROWS PROCESSED ALL TYPES
Number of rows processed in the table space of all types. This figure is meaningful when a table space contains more than one table. Rows for other tables must also be accessed to determine whether they are required to satisfy the request.

ROWS PROCESSED CORRECT TYPE
Number of rows processed that were of the correct type. For a single table space, this should equal ROWS PROCESSED ALL TYPES.

ROWS QUALIFIED BY DATA MANAGER
Number of rows that qualified for the request by DB2's Data Manager component.

ROWS QUALIFIED BY RDS
Number of rows that qualified for the request by DB2's Relational Database Support component.

ROWS INSERTED
Number of rows inserted.

ROWS UPDATED
Number of rows updated.

ROWS DELETED
Number of rows deleted.

PAGES SCANNED
Number of database pages that needed to be scanned to satisfy the request.

REFERENTIAL INTEGRITY PROCESSING:
Data displayed includes:

PAGES SCANNED
Number of pages scanned to enforce referential integrity constraints.

ROWS DELETED/SET NULL
Number of rows deleted or set to a null value to enforce referential integrity.

LOB- PAGES SCANNED
Number of additional pages scanned in a LOB table space.

LOB- PAGES UPDATED
Number of LOB data pages updated by an SQL INSERT or UPDATE.

Stored Procedures Entry Pop-Up (IFCID 233)

Additional data for the STORPROC event is displayed in the Stored Procedures Entry pop-up display, as shown in Figure 135.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    07:50:25  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> NEWIFIS,SEQ=1,LEVEL=2                        ROW 1 OF 5 SCROLL=> CSR

START: 13:51:16 AUTH: BOLSMR2  PLAN: DSNCAL2  CORR: BOLSMRX2     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
STORPROC          0.688                   *MODE=ENTRY PROC=DSNSTO2
================================================================================
         LOCATION NAME: DB2G
         COLLECTION ID: DSN8CAL2
          PROGRAM NAME: DSNCAL2
         PRECOMPILE TS: 1570F6EA083EC9D0
      STATEMENT NUMBER: 60
```

Figure 135. DTRAC Stored Procedures Entry Pop-Up Display

This pop-up display shows the following:

LOCATION NAME
: Location name of the requester

COLLECTION ID
: Package collection ID

PROGRAM NAME
: Caller of the stored procedure

PRECOMPILER TS
: Precompiler timestamp (shown in hex)

STATEMENT NUMBER
: Statement number of the CALL statement in PROGRAM NAME

Stored Procedures Exit Pop-Up (IFCID 233)

Additional data for the STORPROC event is displayed in the Stored Procedures Exit pop-up display, as shown in Figure 136.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    07:50:37  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> NEWIFIS,SEQ=1,LEVEL=2                       ROW 1 OF 11 SCROLL=> CSR

START: 13:51:16 AUTH: BOLSMR2  PLAN: DSNCAL2  CORR: BOLSMRX2     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
STORPROC          2.592                   *MODE=EXIT  PROC=DSNSTO2
================================================================================
         LOCATION NAME: DB2G
         COLLECTION ID: DSN8CAL2
          PROGRAM NAME: DSNCAL2
         PRECOMPILE TS: 1570F6EA083EC9D0
      STATEMENT NUMBER: 60

  --------------------------- SQLCA RETURN DATA ----------------------------
    SQLCODE: -0     ,  SQLSTATE: 00000 ,  SQLERRP: DSN
SQLERRM:
    SQLERRD1: X'00000000'  SQLERRD2: X'00000000'  SQLERRD3: X'00000000'
    SQLERRD4: X'FFFFFFFF'  SQLERRD5: X'00000000'  SQLERRD6: X'00000000'
```

Figure 136. DTRAC Stored Procedures Exit Pop-Up Display

This pop-up display shows the following:

LOCATION NAME
: Location name of the requester

COLLECTION ID
: Package collection ID

PROGRAM NAME
: Caller of the stored procedure

PRECOMPILER TS
: Precompiler timestamp (shown in hex)

STATEMENT NUMBER
: Statement number of the CALL statement in PROGRAM NAME

SQLCA RETURN DATA
: Return code information returned to the program that issued the SQL CALL statement

Synchronous Write Pop-Up (IFCID 008/009)

Additional data for the SYNC-WRITE event is displayed in the Synchronous Write pop-up display, as shown in Figure 137.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    12:35:34  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TRACE1,SEQ=000002,LEVEL=3            ROW     1 OF     2 SCROLL=> CSR

START: 12:26:16 AUTH: BABUSER  PLAN: DSNTIA21 CORR: DB221REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
    SYNC-WRITE   10.052    62 ms   404 us *DB=DEMODB32 TS=T32      BP32
==============================================================================
             RETURN CODE:        0
             PAGES WRITTEN:      1
             TYPE OF WRITE:  CASTOUT
```

Figure 137. DTRAC Synchronous Write Pop-Up Display

This pop-up display shows the following:

RETURN CODE
: Return code from media manager

PAGES WRITTEN
: Number of pages written

TYPE OF WRITE
: Type can be one of the following:

  NORMAL

  CASTOUT

  UNKNOWN

Table Check Constraint Pop-Up (IFCID 305)

Additional data for the TBL-CONSTR event is displayed in the Table Check Constraint pop-up display, as shown in Figure 138.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    08:50:37  INTVL=> 5  LOG=> N  TGT==> DB2G
PARM ==> TBLCONST,SEQ=16,LEVEL=2                      ROW 1 OF 6 SCROLL=> CSR

START: 08:50:00 AUTH: BOLSMR2  PLAN: DSNTEP41 CORR: BOLSMR4D     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
  TBL-CONSTR      0.301                   *DB=DSN8D41A OB=DSN8S41E
================================================================================

                      CHECK CONSTRAINT NAME:    NUMBER
                    CHECK OPERATION PERFORMED:    ENFORCE
                      CHECK OPERATION RESULTS:    REJ
                     FAILED RECORD IDENTIFIER:    00001301
             FIRST 30 CHARS OF FAILED RECORD:    001335  ROGERT  RABBIT A00 -00
```

Figure 138. DTRAC Table Check Constraint Pop-Up Display

This event records the table check constraint activity. The Table Check Constraint pop-up display shows the activities involved in defining, removing, and enforcing a check constraint on a table.

The following fields are displayed depending on the operation being executed:

CHECK CONSTRAINT NAME
: Name of the table check constraint.

CHECK CONSTRAINT TEXT
: Text of the table check constraint, up to 30 characters.

CHECK OPERATION PERFORMED
: Operation that is using the table check constraint function. Possible values are:

  CREATE
  : Check constraint is being defined with a create table operation.

  ALTER ADD
  : Check constraint is being defined with an alter table operation.

  ALTER ADD ENFORCE
  : Check constraint is being enforced during an alter table operation.

  ALTER ADD DROP
  : Check constraint is being removed with an alter table operation.

  ENFORCE
  : Check constraint is being enforced. DB2 is checking that a row does not violate a check constraint.

CHECK OPERATION RESULTS
Result of enforcing the check constraint. Possible values are as follows:

REJ
Check constraint or row was rejected because of a check constraint violation. The operation was not performed.

OK
No check constraint was violated and the operation was successful.

The following two fields are displayed if the CHECK OPERATIONS RESULTS are REJ:

FAILED RECORD IDENTIFIER
Record identifier (RID) for the record that violated the check constraint.

FIRST 30 CHARS OF FAILED RECORD
First 30 characters of the rejected record that violated the check constraint condition.

Temporary Table Pop-Up (IFCID 311)

Additional data for the TEMPTABLE event is displayed in the Temporary Table pop-up display, as shown in Figure 139.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    11:38:24  INTVL=> 5  LOG=> N  TGT==> DB2HSR
PARM ==> NEWSQL,SEQ=2,LEVEL=2                          ROW 1 OF 9 SCROLL=> CSR

START: 10:14:18 AUTH: BOLSMR3  PLAN: RXDB2    CORR: BOLSMR3      CONN: DB2CALL
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
 TEMPTABLE        1.032                   *TABLE=BOLSMR1 TESTGTT4
================================================================================

                        CREATOR:  BOLSMR1
             CURSOR HOLD STATUS:  CURSOR IS NOT HELD
                  WORKFILE TYPE:  TEMPORARY TABLE WORKFILE
                      OPERATION:  DELETE
                    CURSOR NAME:  ABCD
           PACKAGE COLLECTION ID:  RXD2
                     PROGRAM NAME:  RXSEL1M
PKG VERSION: 0000000000000000000000000000000000000000000000000000008925722001
```

Figure 139. DTRAC Temporary Table Pop-Up Display

This event records information about temporary tables. This information applies to instantiation, delete operations, and alter operations on a temporary table.

The pop-up display shows the following:

CREATOR
: Creator of the temporary table.

CURSOR HOLD STATUS
: Status of the cursor hold, which can be one of the following:

HO	Cursor is held after commit
blank	Cursor is not held after commit

WORKFILE TYPE
: Type of work file associated with the temporary table, which can be one of the following:

TT	Temporary table
C	Cursor on temporary table

OPERATION
: Operation that is using the temporary table function, which can be one of the following:

AT	Alter a temporary table
CI	Create an instance of the temporary table in a work file table space
OC	Open a cursor on a temporary table
D	Delete work files for a temporary table
DA	Delete all rows from a temporary table
CC	Close a cursor on a temporary table

CURSOR NAME
: Name of the cursor used for fetch operations.

PACKAGE COLLECTION ID
: Name of the package collection containing the query that uses the temporary table.

PROGRAM NAME
: Name of the program containing the query that uses the temporary table.

PKG VERSION
: Version of the package containing the query that uses the temporary table.

Timeout Pop-Up (IFCID 196)

Additional data for the TIMEOUT event is displayed in the Timeout pop-up display, as shown in Figure 140.

```
BMC SOFTWARE --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    10:24:50  INTVL=> 5  LOG=> N  TGT==> DB2KSR
PARM ==> DET71,SEQ=3,LEVEL=2                          ROW 1 OF 17 SCROLL=> CSR

START: 10:09:50 AUTH: BOLSMR3  PLAN: RXDB2    CORR: BOLSMR3      CONN: DB2CALL
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
  TIMEOUT        33.197                   *DB=DSN8D71A OB=DSN8S71E NUMBER=1
================================================================================
             LOCK TYPE:      DATA PAGE
             RESOURCE NAME:  DB=DSN8D71A   OB=DSN8S71E
             RESOURCE #:     X'00001200'
             HASH VALUE:     X'00018C12'
             FUNCTION:       LOCK
             STATE:          X
             DURATION:       COMMIT
             ZPARM INTERVAL:  30
             INTERVAL COUNT:  1
             REQUESTING WU:  003900A7-0F14A708
      HOLDER CACHED STMT ID: 00000000
      WAITER CACHED STMT ID: 00000000
 ---------------------LOCK HOLDERS / PRIORITY WAITERS -------------------------
 H/W    PLAN     CORRID/LUWID CONNECT  DURATION     STATE MEMBER   OWNER WU
 ------ -------- ------------ -------- ------------ ----- -------- ------------
 HOLDER RXDB2    BOLSMR4      DB2CALL  COMMIT       X     DB2K     00390088-
             USBMCN01LUDB2KB5CF17C8FDB80001      42                0F1488F8
```

Figure 140. DTRAC Timeout Pop-Up Display

This timeout pop-up display shows the following:

LOCK TYPE
: Type of locked resource, such as a data page, row, or page set in a table space or index, or an internal lock to control special processing such as utility serialization, data set open, or altering a buffer pool. See the “Lock Type Table” appendix for a complete list. The long version of the lock type is shown in this pop-up.

RESOURCE NAME
: Specific locked resource. The most common values are DB=database OB-pageset, but there are other possibilities, such as buffer pool ID. See the “Lock Type Table” appendix for a complete list.

RESOURCE #
: Sometimes referred to as the *small resource name*. It consists of a number. For example, for a DATA PAGE lock, a physical page number is shown as X‘hhhhhh00’. An INDEX PAGE lock on the root page is indicated with the special value of X‘000002FF’. A ROW lock is shown as X‘hhhhhhrr’, where the first six digits are the page number and the last two are the RID. For some resources, these numbers are blank. See the “Lock Type Table” appendix for a complete list.

HASH VALUE
: Hash value of the locked resource name.

FUNCTION
Requested function, which can be one of the following:

LOCK function
CHANGE function

STATE
Requested state, which can be one of the following:

UNS UNPROTECTED SHARE
IS INTENT SHARE
IX INTENT EXCLUSIVE
S SHARED
U UPDATE
SIX SHARED INTENT EXCLUSIVE
NSU NON-SHARED UPDATE
X EXCLUSIVE

DURATION
Requested duration, which can be one of the following:

MANUAL
Lock is held only as long as necessary.

MANUAL+1
Temporary change of consistency level from CS to RR during BIND and DDL (RDS changes back internally).

COMMIT
Lock is held until COMMIT.

COMMIT-HOLD
Locks needed to maintain the position of a cursor opened *with hold* are held past COMMIT.

ALLOCATION
Lock is held until deallocation.

PLAN
Lock duration defined when the plan is bound.

ZPARM INTERVAL
Resource timeout interval specified during installation (IRLMWT in DSN6SPRM), or the timeout interval that a stored procedure call can wait for a TCB (STORTIME in DSN6SYSP). (DB2 4.1 or higher)

INTERVAL COUNT
Number of timeout intervals that can occur before the agent is timed out. For example, a utility waits the specified number of intervals before timeout. (DB2 4.1 or higher)

REQUESTING WU
Requested owning work unit.

HOLDER/WAITER CACHED STMT ID
SQL statement unique identifier for both the lock holder and the waiter. This ID is a generated number for uniquely identifying a statement in the prepared statement cache.

LOCK HOLDERS/PRIORITY WAITERS

Each agent that caused the timeout is identified in this section. This is normally a holder of an incompatible lock on the resource requested by the timed-out thread. In DB2 4.1, this agent may also be a higher priority waiter.

H/W
: HOLDER for a lock holder. WAITER for a higher priority waiter. (DB2 4.1 or higher)

PLAN
: Holder's plan name.

CORRID/LUWID
: Holder's correlation ID/holder's logical unit of work ID.

CONNECT
: Holder's connection ID.

DURATION
: Holders lock duration, which can be one of the following:

  MANUAL
  : Lock is held only as long as necessary.

  MANUAL+1
  : Temporary change of consistency level from CS to RR during BIND and DDL (RDS changes back internally).

  COMMIT
  : Lock is held until COMMIT.

  COMMIT-HOLD
  : Locks needed to maintain the position of a cursor opened *with hold* are held past COMMIT.

  ALLOCATION
  : Lock is held until deallocation.

  PLAN
  : Lock duration defined when the plan is bound.

STATE
: State of the incompatible lock, which can be one of the following:

UNS	UNPROTECTED SHARE
IS	INTENT SHARE
IX	INTENT EXCLUSIVE
S	SHARED
U	UPDATE
SIX	SHARED INTENT EXCLUSIVE
NSU	NON-SHARED UPDATE
X	EXCLUSIVE

MEMBER
: Name of the DB2 member of a data sharing group (DB2 4.1 or higher).

OWNER WU
: Holder's owning work unit.

Trigger Entry Pop-Up (IFCID 325)

Additional data for the TRIGGER event is displayed in the Trigger Entry pop-up display, as shown in Figure 141.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    12:56:35  INTVL=> 5  LOG=> N  TGT==> DB2JSR
PARM ==> DET,SEQ=5,LEVEL=2                            ROW 1 OF 9 SCROLL=> CSR

START: 08:44:52 AUTH: BOLSMR4  PLAN: SMRTEP6A CORR: BOLSMR5V     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
  TRIGGER         2.561                   * MODE=ENTRY  TRIG=MALE_EMP
================================================================================
     COLLECTION ID:  SMRTEP2A
   PROGRAM/PACKAGE:  DSN@EP2L
  STATEMENT NUMBER:  989
       SCHEMA NAME:  BOLSMR4
         TIMESTAMP:  165919750F007219
   ACTIVATION TIME:  AFTER
       GRANULARITY:  ROW
      SQL STATEMENT:  INSERT
  ACTION CONDITION:  NONE
```

Figure 141. DTRAC Trigger Entry Pop-Up Display

This pop-up display shows the following:

COLLECTION ID
: Collection ID of the package that executed the statement that activated the trigger.

PROGRAM/PACKAGE
: Name of the program or package that executed the statement that activated the trigger.

STATEMENT NUMBER
: Statement number of the SQL statement that activated the trigger. This field is blank if this is a cascaded trigger.

SCHEMA NAME
: Schema name of the trigger.

TIMESTAMP
: Trigger timestamp.

ACTIVATION TIME
: Trigger activation time, which can be one of the following:

  BEFORE
  AFTER

GRANULARITY
: Trigger granularity, which can be one of the following:

ROW	For each row
STATEMENT	For each statement

SQL STATEMENT
Type of triggering SQL statement, which can be one of the following:

UPDATE
INSERT
DELETE

ACTION CONDITION
Triggered action condition evaluation, which can be one of the following:

TRUE	The triggered action condition is true.
FALSE	The triggered action condition is false.
NONE	There is no triggered action condition for the trigger.

Trigger Exit Pop-Up (IFCID 325)

Additional data for the TRIGGER event is displayed in the Trigger Exit pop-up display, as shown in Figure 142.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    12:56:41  INTVL=> 5  LOG=> N  TGT==> DB2JSR
PARM ==> DET,SEQ=5,LEVEL=2                           ROW 1 OF 15 SCROLL=> CSR

START: 08:44:52 AUTH: BOLSMR4  PLAN: SMRTEP6A CORR: BOLSMR5V     CONN: BATCH
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 TRIGGER          2.580                   * MODE=EXIT   TRIG=MALE_EMP
================================================================================
     COLLECTION ID:  SMRTEP2A
   PROGRAM/PACKAGE:  DSN@EP2L
  STATEMENT NUMBER:  989
       SCHEMA NAME:  BOLSMR4
         TIMESTAMP:  165919750F007219
   ACTIVATION TIME:  AFTER
       GRANULARITY:  ROW
     SQL STATEMENT:  INSERT
  ACTION CONDITION:  FALSE

  -------------------------- SQLCA RETURN DATA ----------------------------
    SQLCODE: -0     ,  SQLSTATE: 00000 ,  SQLERRP: DSN
SQLERRM:
    SQLERRD1: X'00000000'  SQLERRD2: X'00000000'  SQLERRD3: X'00000000'
    SQLERRD4: X'00000000'  SQLERRD5: X'00000000'  SQLERRD6: X'00000000'
```

Figure 142. DTRAC Trigger Exit Pop-Up Display

This pop-up display shows the following:

COLLECTION ID
: Collection ID of the package that executed the statement that activated the trigger.

PROGRAM/PACKAGE
: Name of the program or package that executed the statement that activated the trigger.

STATEMENT NUMBER
: Statement number of the SQL statement that activated the trigger. This field is blank if this is a cascaded trigger.

SCHEMA NAME
: Schema name of the trigger.

TIMESTAMP
: Trigger timestamp.

ACTIVATION TIME
: Trigger activation time, which can be one of the following:

  BEFORE
  AFTER

GRANULARITY
Trigger granularity, which can be one of the following:

ROW	For each row
STATEMENT	For each statement

SQL STATEMENT
Type of triggering SQL statement, which can be one of the following:

UPDATE
INSERT
DELETE

ACTION CONDITION
Triggered action condition evaluation, which can be one of the following:

TRUE	The triggered action condition is true.
FALSE	The triggered action condition is false.
NONE	There is no triggered action condition for the trigger.

SQLCA RETURN DATA
Return code information returned to the program that executed the statement that activated the trigger.

User-Defined Function Pop-Up (IFCID 324)

Additional data for the UDF event is displayed in the User-Defined Function pop-up display, as shown in Figure 143.

```
BMC Software --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    14:39:55  INTVL=> 5  LOG=> N  TGT==> DB2JSR
PARM ==> DTRAC,SEQ=4,LEVEL=2                          ROW 1 OF 23 SCROLL=> CSR

START: 14:35:45 AUTH: BOLSMR4  PLAN: SMRTEP6A CORR: BOLSMR2V     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
 UDF              0.005                    *APPL=         PLAN=SMRTEP6A
==============================================================================
 QUERY NUMBER:       1,361
 GROUP MEMBER:       DB2J
 COLLECTION ID:      SMRTEP2A
 PROGRAM NAME:       DSN@EP2L
 CONSISTENCY TOKEN:  1628B4B6C4C2F2D1
 TIMESTAMP:          20000125C4C2F2D100004040
 CURRENT PATH:       "SYSIBM","SYSFUN","SYSPROC","BOLSMR4"




 **** FUNCTION INFORMATION ****
 QUERY BLOCK NUM:    1
 FUNCTION SCHEMA:    DSN8
 FUNCTION NAME:      ALTDATE
 SPECIFIC FUNCTION:  DSN8DUADV
 FUNCTION TYPE:      SCALAR UDF
 VIEW CREATOR:
 VIEW NAME:
 FUNCTION TEXT:      DSN8.ALTDATE( 'DD MONTH YY' )
```

Figure 143. DTRAC User-Defined Function Pop-Up Display

This pop-up display shows the following:

QUERY NUMBER
: Number of the query that the user-defined function is executing.

GROUP MEMBER
: Name of the member of the data sharing group under which the user-defined function is currently executing.

COLLECTION ID
: Collection ID of the package that invoked the user-defined function.

PROGRAM NAME
: Name of the program that invoked the user-defined function.

CONSISTENCY TOKEN
: Consistency token for the user-defined function.

TIMESTAMP
: Bind time of the user-defined function.

CURRENT PATH
Value of the current path special register under which the user-defined function is currently executing.

QUERY BLOCK NUM
Number of the query block.

FUNCTION SCHEMA
Schema name of the user-defined function.

FUNCTION NAME
Name of the user-defined function.

SPECIFIC FUNCTION
Specific name of the user-defined function.

FUNCTION TYPE
Type of user-defined function, which can be one of the following:

SCALAR UDF	User-defined scalar function
TABLE UDF	User-defined table function

VIEW CREATOR
Name of the view creator, if the user-defined function is referenced in a view definition.

VIEW NAME
Name of the view, if the user-defined function is referenced in a view definition.

FUNCTION TEXT
Up to the first 254 characters of the function invocation, including the function name and its parameters.

Utility Pop-Up

Additional data for utility events, such as RUNSTATS or RECOVER, is displayed in the Utility pop-up display, as shown in Figure 144.

```
BMC SOFTWARE --------------   DETAIL TRACE ENTRY    -------------  RX AVAILABLE
SERV ==> DTRAC           INPUT    08:52:41  INTVL=> 5  LOG=> N  TGT==> DB2KSR
PARM ==> DET2,SEQ=26,LEVEL=2                          ROW 1 OF 5 SCROLL=> CSR

START: 08:37:19 AUTH: BOLSMR4  PLAN: DSNUTIL  CORR: DMRREQ71     CONN: UTILITY
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
REORG             0.233    60 ms 3,407 us *ID=DSNTEX               PH=UTILINIT
================================================================================
         RECORDS PROCESSED:      0
                  DATABASE: DEMOD71A
                    OBJECT: SE71SD
                DS OR PART#:        0
                 KEYWORD(S): FASTSWITCH
```

Figure 144. DTRAC Utility Pop-Up Display

This pop-up display shows the following:

xxxxxxx PROCESSED
: Type of item processed by the utility, which can be PAGES, RECORDS, or OBJECTS.

DATABASE
: Database name.

OBJECT
: Table space or index name.

DS OR PART#
: Partition or data set number, if the utility is operating on a single partition or data set (DB2 version 5 and later).

KEYWORD(S)

Utility keyword invocation field.

For the COPY utility, it can be one of the following:

CONCURRENT
SHRLEVEL CHANGE
PARALLEL
CHECKPAGE

For the RECOVER utility, it can be one of the following:

REUSE
PARALLEL

For the LOAD utility, it can be one of the following:

KEEPDICTIONARY
REUSE
LOG NO
SORTKEYS
SHRLEVEL CHANGE
COPYDDN
STATISTICS
PART INDDN

For the REORG utility, it can be one of the following:

KEEPDICTIONARY
REUSE
LOG NO
SORTKEYS
SORTDATA
NOSYSRED
SHRLEVEL CHANGE
SHRLEVEL REFERENCE
COPYDDN
STATISTICS
FASTSWITCH

For the REBUILD utility, it can be one of the following:

REUSE
SORTKEYS
STATISTICS
WORKDDN

For the RUNSTATS utility, it can be one of the following:

SAMPLE
SHRLEVEL

For the UNLOAD utility, it can be one of the following:

SHRLEVEL REFERENCE
SHRLEVEL CHANGE ISOLATION CS
SHRLEVEL CHANGE ISOLATION UR

VTAM Exit Pop-Up (IFCID 164)

Additional data for the VTAM-EXIT event is displayed in the VTAM Exit pop-up display, as shown in Figure 145.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:22:56  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=3                                ROW 1 OF 6 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
===============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
  VTAM-EXIT     150.050                   *TYPE=RPL
===============================================================================

                  LUNAME:             LUDB1D
                  MODE:               IBMDB2LM
                  CONVERSATION LIMIT: 20
                  CONVERSATION ID:    X'01000007'
                  INSTANCE ID:        X'F09F01ADDBD13AFB'
```

Figure 145. DTRAC VTAM Exit Pop-Up Display

This pop-up display shows the following:

LUNAME
: VTAM logical unit name

MODE
: VTAM mode name

CONVERSATION LIMIT
: Maximum number of conversations

CONVERSATION ID
: Four-byte logical connection identifier in hexadecimal

INSTANCE ID
: Eight-byte VTAM session instance identifier in hexadecimal

VTAM Macro Pop-Up (IFCID 165)

Additional data for the VTAM-MACRO event is displayed in the VTAM Macro pop-up display, as shown in Figure 146.

```
BMC Software --------------   DETAIL TRACE ENTRY    -----------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    09:21:54  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> ,SEQ=2,LEVEL=3                                ROW 1 OF 5 SCROLL=> CSR

START: 09:11:43 AUTH: CIR2X    PLAN: DSNESPRR CORR: CIR2X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -----------------------------------
  VTAM-MACRO      8.139                   *RC(   0) APPCCMD  PREPRCV  DATAFLU
================================================================================

                  FDBK2:           X'00'
                  RCPRI:           X'0000'
                  RCSEC:           X'0000'
                  CONVERSATION ID: X'01000007'
```

Figure 146. DTRAC VTAM Macro Pop-Up Display

This pop-up display shows the following:

FDBK2
: Reason code from VTAM

RCPRI
: Primary return code from VTAM

RCSEC
: Secondary return code from VTAM

CONVERSATION ID
: Four-byte logical connection identifier in hexadecimal

Wait I/O Pop-Up (IFCID 127/128)

Additional data for the WAIT-I/O event is displayed in the Wait I/O pop-up display, as shown in Figure 147.

```
BMC Software --------------  DETAIL TRACE ENTRY ---------------PERFORMANCE MGMT
SERV ==> DTRAC           INPUT    12:35:34  INTVL=> 3  LOG=> N  TGT==> DB2C
PARM ==> TRACE1,SEQ=000002,LEVEL=3            ROW     1 OF     2 SCROLL=> CSR

START: 12:26:16 AUTH: BABUSER  PLAN: DSNTIA21 CORR: DB221REQ     CONN: BATCH
==============================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
    WAIT-I/O     10.052    62 ms   404 us *DB=DEMODB32 TS=T32      BP32 SYNC-RD
==============================================================================
             PAGE NUMBER:   795
             (IO REQUESTOR CANCELLED)
```

Figure 147. DTRAC Wait I/O Pop-Up Display

This pop-up display shows the following:

PAGE NUMBER
: Relative page number of the page your request is waiting for

IO REQUESTOR CANCELLED
: Appears only if the condition is true

UTRAC—User Detail Trace

The UTRAC User Detail Trace display shows all the traced events of the current active thread for a specified user. It can be accessed only for threads that are being traced (see "Starting a Detail Trace" on page 21).

```
BMC Software --------------   USER DETAIL TRACE     -----------PERFORMANCE MGMT
SERV ==> UTRAC           INPUT    17:25:19  INTVL=> 8  LOG=> N  TGT==> DB2D
PARM ==> CIR4X,SEQ=000001,LEVEL=2             ROW     1 OF    85 SCROLL=> CSR
EXPAND:  LINESEL(DETAIL)
START: 17:21:38 AUTH: CIR4X    PLAN: DSNESPRR CORR: CIR4X        CONN: TSO
================================================================================
    EVENT        AT     ELAPSED     CPU   DETAIL
-------------- -------- -------- -------- -------------------------------------
SYNC              0.000 4,049 us 1,619 us
 LOCK-SUMMARY     0.002                   *MAXPG(12) ESCL(0) TS(3)
PREPARE    71   115.047   747 ms    37 ms *RC(   0) C=C1         D/X   PS(  22)
 BIND-TEXT      115.052                   *TYPE=DYNAMIC  TEXT=SELECT DBID, OBI+
 EXPLAIN        115.791                   *PLAN=DSNESPRR COST=0.0
OPEN      131   115.797 2,705 ms    87 ms *RC(   0) C=C1         D/X/W PS( 707)
 SORT           115.797 2,703 ms    86 ms  REC(    78) RECSZ(   26) WKFILE(  2)
FETCH     125   118.504 1,235 us   664 us *RC(   0) C=C1         W     PS(   2)
FETCH     125   118.564   263 us   263 us *RC(   0) C=C1         W
FETCH     125   118.565   291 us   292 us *RC(   0) C=C1         W
FETCH     125   118.566   232 us   233 us *RC(   0) C=C1         W
FETCH     125   118.567   230 us   230 us *RC(   0) C=C1         W
FETCH     125   118.569   257 us   257 us *RC(   0) C=C1         W
FETCH     125   118.570   231 us   231 us *RC(   0) C=C1         W
FETCH     125   118.571   361 us   258 us *RC(   0) C=C1         W
FETCH     125   118.572   332 us   248 us *RC(   0) C=C1         W
```

Figure 148. User Detail Trace Panel

Select Code:
UTRAC

Parameter:
The first parameter for this panel is required to identify the thread to be displayed. Any of the following identifiers can be used:

User ID
Correlation ID
Authorization ID

The following parameters are optional:

LEVEL=1|2|3
The default is 2. Levels 1 or 3 can be typed in this field to change the display at any time. The entries shown at the three levels depend upon the type of detail trace started (see Appendix B, "IFCID Table (ATRAC Data Collection)" on page 409 and "Event Data" on page 259).

Note: There are different levels of data collection started with an ATRAC request. This means the correct type of trace must be started to see all the events. A warning message is displayed on the fourth line of the display if all possible data is not available.

SEQ=
Unique internal sequence number assigned to each trace entry. When UTRAC is first selected, the first active entry in the trace buffer is displayed and identified in the SEQ field. Any valid trace buffer entry can be displayed by typing its trace sequence number in the SEQ field.

Description:
Shows all the traced events of the current active thread for the specified user. It clears when the thread terminates.

The traced events can be displayed at three succeeding levels by using the LEVEL= parameter. These display levels provide the following:

LEVEL=1 Displays only those events related to user requests (SQL statements) plus any exception conditions such as authorization failures, timeouts, or deadlocks.

LEVEL=2 Displays level 1 events plus events related to DB2 requests such as CREATE THREAD, EDM requests, or sorts. This display is the default and is the initial display shown when UTRAC is invoked.

LEVEL=3 Displays level 1 and level 2 events plus events related to DB2 processing caused by the user and DB2 requests such as scan, I/O, and lock/latch activity.

Scrolling:
Each display can be scrolled up (PF7/19) or down (PF8/20). Data for only the current entry in the trace buffer is scrolled. A new transaction in the trace buffer can be displayed by using the left and right PF scroll keys. PF10/22 (left) moves backwards through the transaction entries in the buffer. PF11/23 (right) moves forwards through the transaction entries. The transaction displayed is identified in the SEQ field of the display.

Expand:
Pop-up displays of additional statistical information can be selected by positioning the cursor anywhere on the line of an entry that has an asterisk (*) in the DETAIL column and pressing ENTER.

An asterisk (*) preceding a field in the DETAIL column means a pop-up display is available for that event.

Display Fields

The display fields on the UTRAC panel are the same as those for the DTRAC display. For a detailed description of these fields, see “Display Fields” on page 258 and “Event Pop-Up Displays” on page 274.

Chapter 15. Viewing I/O Activity

To view I/O activity, expand from LTRAC or TSTAT to the DBIO display. From there, you can expand to the other DBIOx displays described in this chapter.

DBIOx—I/O Analysis (Trace)

The I/O Analysis (Trace) panels display information about I/O activity that is collected in a detail trace that includes I/O events. When viewing an I/O trace, an expand button of I/O—DB/TS is displayed on the LTRAC list of completed threads and also on the TSTAT summary display.

A traced I/O event identifies not only the table space accessed, its buffer pool, timestamp and elapsed time, but also the thread that caused the I/O to occur. This allows analysis of various causes of I/O problems, such as:

- Poor I/O response (data set / volume level)
- Excessive I/O caused by an application
- Contention at particular time periods between applications for buffer space or data on the same volumes

I/O events can be summarized by the following:

- Database and table space (DBIO service)
- Plan (DBIOP service)
- Authorization ID (DBIOA service)
- Buffer pool (DBIOF service)
- Time (DBIOT service - selectable interval of minutes or hours)
- Connection name (DBIOC service)
- Location (DBIOL service)

If the detail trace also includes SQL events, two additional summary displays are available:

- Package/Program (DBIOK service)
- SQL statement (DBIOS service)

The summary by database and table space (DBIO) can be cursor-selected from the EXPAND line of the LTRAC display service. The others can be cursor-selected from the EXPAND line of the DBIO display.

The requests can be for a specific purpose or situation. You can use the trace I/O displays as you need them to analyze I/O activity over time, by application, or by user instead of waiting for batch reports. They can be used to view both current traces and history traces.

After summarizing by one of these other categories, you can sort the display by any column and then cursor-select a single line in that display; for example, the plan with the highest I/O count. This returns you to a list of table spaces, but now it is restricted to only the selected identifier; for example, for a single plan or time interval. This process can be repeated until you have narrowed the focus to a specific problem area.

Display Fields

The columns for all I/O analysis displays include the values shown in Figure 149.

```
BOOLE & BABBAGE ----------- I/O ANALYSIS BY xxxxxxx  ------------------------
SERV ==> DBIOx           INPUT    14:27:17  INTVL=> 3  LOG=> N  TGT==> DB2D
PARM ==> traceid,TOTAL,SORT=xx                     ROW 1 OF 93 SCROLL  => CSR
EXPAND:  LINESEL(DBIO), CATALOG, HISTORY
OPTION:  TOTAL, SYNC, ASYNC

07FEB94                  X----TRACE TITLE-------X
                   sum-     I/O    I/O    MAX     AVG
sumkey1  sumkey2  key3    COUNT    %   IOWAIT  IOWAIT
-------- -------- ---- -------- -----  --ms--  --ms-- 0 ...20...40...60...80ms
xxxxxxxx xxxxxxxx      xxxxxxxx xxx.x  xxxxxx  xxxxxx | ********             |
```

Figure 149. I/O Analysis Display Fields

An I/O analysis display shows the summarized I/O events for each object in a row. The entries are shown by the following:

summary key
: There can be 1–3 summary keys, as follows:

- DATA BASE and TABLE SPACE (DBIO)
- AUTHID (DBIOA)
- BUFFER POOL (DBIOF)
- CONNECT (DBIOC)
- INTERVAL START (DBIOT)
- LOCATION (DBIOL)
- PLAN (DBIOP)
- PLAN and PACKAGE/PROGRAM (DBIOK) (only with SQL trace)
- PLAN and PACKAGE/PROGRAM and SQL STMT (DBIOS) (only with SQL trace)

ddmmmyy
: Date I/O events were collected (earliest if more than one).

Title
: Trace title.

I/O COUNT
: Number of I/Os to this object, total or only synchronous or asynchronous, depending on the parameter.

I/O %
: Percent of total I/Os issued to this object.

Note: The total is the sum of the I/Os for all objects in the scrollable display, not for all monitored objects. For example, if parameters of ,SYNC,BP=BP1 restrict the display to only synchronous I/O events to the table spaces in BP1, the total is also only for these table spaces.

MAX IOWAIT
: Highest I/O wait time observed for this object. A high value here may indicate a potential problem that is masked in the average value.

AVG IOWAIT
Average I/O wait time observed for this object.

Note: Both the maximum and average wait times are those observed by DB2, so this is the actual delay for the application. It may be higher than the DASD volume response time, since it can include MVS delays in redispatching the task.

Graph of Average I/O Wait Time
Graph of average I/O wait time is based on a constant maximum value, so that the scale does not change when moving from one display to another. The default values are 40ms for synchronous I/O and 80ms for asynchronous and total I/O.

Description

Displays I/O information about table spaces, buffer pools, or workload. It includes a count of the number of I/Os processed, the percent of the total I/Os for the selected objects, and the maximum and average I/O wait time. This information is used to analyze how I/O is balanced across table spaces or buffer pools, which applications are causing the most I/O, and to detect I/O response time problems.

This list is scrollable.

Service Message(s)

A message in the parameter field of the service shows the sequence number of the first object displayed and the total number of objects that can be displayed in the scrollable list; for example:

```
ROW 1 OF 85
```

Using EXPAND

All the DBIOx services can be EXPANDed to the following displays:

LINESEL(DBIO)
Display of I/O data by table space for the selected summary key.

Position the cursor anywhere in a summary line for a summary key entry and press ENTER. PF3 (END) from that display returns to this list.

CATALOG
Corresponding RxD2 catalog display for the object(s) selected in the DBTS= parameter if RxD2 is installed. A detailed list of databases, table spaces, or indexes is shown.

Detailed catalog information is available per object from these lists. They provide access to tablepart or indexpart specifications, as well as a LISTCAT option.

Note: Access is possible to remote DB2 subsystems only if connected with DDF to a local DB2 (in the same MVS system as the TS).

HISTORY
Shows the data from the current trace log data set for this trace (if you are viewing a current trace with logging, this provides access to data no longer in the online buffer). It is only shown when logging is active.

Sorting the Display

The I/O analysis displays show a list of scrollable, summarized I/O entries sorted by one or more summary keys (first display columns). Any of the column data in the display can be sorted by typing the first two characters of the column heading with the SORT parameter as

```
,SORT=cc
```

where cc can be any of the following two characters:

DB|AU|BP|CO|LO|PL|IS|ISD
: Sorts the list alphabetically by the summary key(s):

- Database and table space (DATABASE and TABLE SPACE columns in DBIO service)
- Authorization ID (AUTHID column in DBIOA service)
- Buffer pool (BUFFER POOL column in DBIOF service)
- Connection name (CONNECT column in DBIOC service)
- Location name (LOCATION column in DBIOL service)
- Plan name (PLAN column in DBIOP service)
- Plan name (PLAN and PACKAGE columns in DBIOK service)
- Plan name (PLAN and PACKAGE and SQL columns in DBIOS service)
- Ascending time (INTERVAL START column in DBIOT service)
- Descending time (INTERVAL START column in DBIOT service)

IC
: Sorts the list in a numerically descending order (I/O COUNT column)

I%
: Sorts the list in a numerically descending order (I/O % column)

MI
: Sorts the list in a numerically descending order (MAX IOWAIT column)

AI
: Sorts the list in a numerically descending order (AVG IOWAIT column)

Note: You also can sort on each column by tabbing to the column header and pressing ENTER.

Selecting a Subset of I/O Events

The following selection keywords can be typed directly in the parameter field:

,DBTS=(dbname,tsname)
: Selects only the partitions / data sets in the specified table space. Both dbname and tsname are required; however, generic qualifiers can be used to specify them. See “Qualifying Requests” in *Using MAINVIEW.*

,A|AUTHID=xxxxxxxx
: Selects a subset of the I/O events by authorization ID. Specify one authorization ID or a generic group of authorization IDs by replacing character positions with a + (one position) or an * (multiple positions).

,BP=BPnn
Selects only those I/O events for the specified buffer pool.

,C|CONNECT=xxxxxxxx
Selects a subset of the I/O events by connect type. Specify one connect type (TSO, BATCH, DB2CALL, imsid, cicsjobname) or a generic group of connections by replacing character positions with a + (one position) or an * (multiple positions).

,L|LOC=xxxxxxxx
Selects a subset of the I/O events by location. Specify one location or a generic group of locations by replacing character positions with a + (one position) or an * (multiple positions).

,P|PLAN=xxxxxxxx
Selects a subset of the I/O events by plan. Specify one plan or a generic group of plans by replacing character positions with a + (one position) or an * (multiple positions).

,T|TIME=hhmm[-hhmm]
Selects a subset of the I/O events by a start time or time period, where hhmm indicates hours and minutes.

Midnight wrap is supported, where the start time is higher than the end time; for example, TIME=2000-3000. The end timestamp of an accounting record is used to assign it a time period.

,D|DAY=nn[-nn]
Selects a subset of the I/O events by a day or range of days, where nn can be a relative number from the date of the first accounting record in the trace buffer (first date = 1, up to a maximum of 99). If a time period is selected without a DAY specification, it defaults to the day of the most current record in the buffer.

Note: Each keyword can be specified only once. If more than one keyword is specified, the result is ANDed.

The parameter field is filled in automatically with any of these keywords when a series of summarizations and selections are made; for example:

- From DBIO, select PLAN (DBIOP—I/O Analysis by Plan) to see all I/O events summarized by plan.
- From DBIOP, sort by percent of total I/Os (SORT=I%) and cursor select (expand) the line for the plan with the highest percent of I/Os. This returns to DBIO but displays only the table spaces and I/O for that one plan. The parameter field is filled in with PLAN= xxxxxxxx.
- From this DBIO subset, select TIME (DBIOT—I/O Analysis by Time) to see a summary of all the plan I/O activity by time interval. The default interval is one hour, but this can be modified to any interval of minutes or hours. The parameter field is still filled in with PLAN=xxxxxxxx.

DBIO—I/O Analysis by Database / Table Space (DB/TS)

The I/O Analysis by DB/TS panel is a scrollable list of I/O events summarized and sorted by database and table space (DATABASE and TABLESPACE columns).

```
BMC Software --------------    I/O Analysis-HT      -----------PERFORMANCE MGMT
SERV ==> DBIO            INPUT    11:02:14  INTVL=> 3  LOG=> N  TGT==> DB2F
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

19MAR01                    CIR2.DB2F.JQSIO.MAR19.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  DSNSCT02             1   1.4     28     28 | *******              |
DSNDB01  DSNSPT01             1   1.4     17     17 | ****                 |
DSNDB01  SCT02                2   2.8     38     31 | *******              |
DSNDB01  SPT01                4   5.6    351     95 | ******************** |
```

Select Code:

DBIO

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=DB|IC|I%|MI|AI
: When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by database and table space (default) and ,SORT=DB is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters
: "Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

DATABASE is the first column and TABLESPACE is the second column of this display. They are the two summary keys used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by database and table space. The display is a scrollable list of summary lines for each unique table space found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

Expand

The DBIO display can be EXPANDed to the following displays (in addition to those described in "Using EXPAND" on page 385):

PLAN
: I/O Analysis by Plan display (DBIOP) that summarizes the I/O events from this trace by plan.

AUTH
: I/O Analysis by AUTHID display (DBIOA) that summarizes the I/O events from this trace by authorization ID.

BPOOL
: I/O Analysis by BPOOL display (DBIOB) that summarizes the I/O events from this trace by buffer pool.

TIME
: I/O Analysis by Time display (DBIOT) that summarizes the I/O events from this trace by time interval.

CONNECT
: I/O Analysis by Connect display (DBIOC) that summarizes the I/O events from this trace by connection name.

LOCATION
: I/O Analysis by Location display (DBIOL) that summarizes the I/O events from this trace by location name.

LINESEL(DBTS)
: Status display of open databases and table spaces, which includes the first volume and number of extents per data set, showing only the data sets or partitions within the selected table space.

CATALOG
: Corresponding RxD2 catalog display for the object(s) selected in the DBTS= parameter if RxD2 is installed. A detailed list of databases, table spaces, or indexes is shown.

  Detailed catalog information is available per object from these lists. They provide access to tablepart or indexpart specifications, as well as a LISTCAT option.

  Note: Access is possible to remote DB2 subsystems only if connected with DDF to a local DB2 (in the same MVS system as the TS).

DBIOA—I/O Analysis by Authorization ID (AUTHID)

The I/O Analysis by AUTHID panel is a scrollable list of I/O events summarized and sorted by authorization ID (AUTHID column).

```
BMC Software --------------  I/O Analysis-Authid    -----------PERFORMANCE MGMT
SERV ==> DBIOA           INPUT    11:04:22  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=AU,TOTAL                          LINE    1 OF    1 SCROLL=> CSR
EXPAND:  LINESEL(DBIO), CATALOG
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    CIR2.DB2F.JQSIO.MAR19.T1818.V01
                           I/O   I/O   MAX    AVG
AUTHID                    COUNT    %  IOWAIT IOWAIT
--------               -------- ----- --ms-- --ms-- 0 ...20...40...60...80
CIR1                         71   100   1028     51 | ************         |
```

Select Code:

DBIOA

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=AU|IC|I%|MI|AI
: When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by authorization ID (default) and ,SORT=AU is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters
: "Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

AUTHID is the first column of this display and is the summary key used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by authorization ID. The display is a scrollable list of summary lines for each unique authorization ID found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

DBIOF—I/O Analysis by Buffer Pool (BPOOL)

The I/O Analysis by BPOOL panel is a scrollable list of I/O events summarized and sorted by buffer pool (BUFFER POOL column).

```
BMC Software --------------   I/O Analysis-Bpool    -----------PERFORMANCE MGMT
SERV ==> DBIOF           INPUT    11:20:33  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=BP,TOTAL                          LINE    1 OF    2 SCROLL=> CSR
EXPAND:  LINESEL(DBIO), CATALOG
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    CIR2.DB2F.JQSIO.MAR19.T1818.V01
BUFFER                     I/O   I/O   MAX    AVG
POOL                      COUNT    %  IOWAIT IOWAIT
--------               -------- ----- --ms-- --ms-- 0 ...20...40...60...80
BP0                          67  94.4   1028     52 | *************        |
BP32K                         4   5.6     37     23 | *****                |
```

Select Code:

DBIOF

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=BP|IC|I%|MI|AI
: When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by buffer pool (default) and ,SORT=BP is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters
: "Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

BUFFER POOL is the first column of this display and is the summary key used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by buffer pool. ID. The display is a scrollable list of summary lines for each unique buffer pool found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

DBIOC—I/O Analysis by Connection Name (CONNECT)

The I/O Analysis by CONNECT panel is a scrollable list of I/O events summarized and sorted by connection name (CONNECT column).

```
BMC Software --------------  I/O Analysis-Connect   -----------PERFORMANCE MGMT
SERV ==> DBIOC           INPUT    11:27:55  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=CO,TOTAL                          LINE    1 OF    2 SCROLL=> CSR
EXPAND:  LINESEL(DBIO), CATALOG
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    CIR2.DB2F.JQSIO.MAR19.T1818.V01
                           I/O   I/O   MAX    AVG
CONNECT                   COUNT    %  IOWAIT IOWAIT
--------               -------- ----- --ms-- --ms-- 0 ...20...40...60...80
BATCH                        35  49.3     89     20 | *****                |
DB2CALL                      36  50.7   1028     80 | ******************** |
```

Select Code:

DBIOC

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=CO|IC|I%|MI|AI
: When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by connection name (default) and ,SORT=CO is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters
: "Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

CONNECT is the first column of this display and is the summary key used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by connection name. The display is a scrollable list of summary lines for each unique connection name found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

DBIOL—I/O Analysis by Location (LOCATION)

The I/O Analysis by LOCATION panel is a scrollable list of I/O events summarized and sorted by location (LOCATION column).

```
BMC Software --------------    I/O Analysis-Loc     -----------PERFORMANCE MGMT
SERV ==> DBIOL           INPUT    11:28:32  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=LO,TOTAL                          LINE    1 OF    2 SCROLL=> CSR
EXPAND:  LINESEL(DBIO), CATALOG
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    CIR2.DB2F.JQSIO.MAR19.T1818.V01
                           I/O   I/O   MAX    AVG
LOCATION                  COUNT    %  IOWAIT IOWAIT
-----------------      -------- ----- --ms-- --ms-- 0 ...20...40...60...80
DB2E                          18  25.4     40     23 | *****                |
DB2F                          53  74.6   1028     60 | ***************      |
```

Select Code:

DBIOL

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=LO|IC|I%|MI|AI
: When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by location name (default) and ,SORT=LO is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters
: "Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

LOCATION is the first column of this display and is the summary key used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by location name. The display is a scrollable list of summary lines for each unique location found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

DBIOP—I/O Analysis by Plan (PLAN)

The I/O Analysis by PLAN panel is a scrollable list of I/O events summarized and sorted by plan (PLAN column).

```
BMC Software --------------   I/O Analysis-Plan     -----------PERFORMANCE MGMT
SERV ==> DBIOP           INPUT    11:29:20  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=PL,TOTAL                          LINE    1 OF    3 SCROLL=> CSR
EXPAND:  PKG/PGM, LINESEL(DBIO), CATALOG
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    CIR2.DB2F.JQSIO.MAR19.T1818.V01
                           I/O   I/O   MAX    AVG
PLAN                      COUNT    %  IOWAIT IOWAIT
--------               -------- ----- --ms-- --ms-- 0 ...20...40...60...80
DSNTIA31                     29  40.8     89     19 | ****                 |
DSNTIB31                      6   8.5     38     25 | ******               |
RXDB2                        36  50.7   1028     80 | ******************** |
```

Select Code:

DBIOP

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=PL|IC|I%|MI|AI
: When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by plan (default) and ,SORT=PL is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters
: "Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

PLAN is the first column of this display and is the summary key used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by plan. The display is a scrollable list of summary lines for each unique plan found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

Expand:

Can also EXPAND to the following display (in addition to those described in "Using EXPAND" on page 385):

PKG/PGM

Display of I/O events summarized and sorted by plan and package/program name (DBIOK)

DBIOK—I/O Analysis by Package / Program (PKG/PGM)

The I/O Analysis by PKG/PGM panel is a scrollable list of I/O events summarized and sorted by plan and package/program name (PLAN and PACKAGE/PROGRAM columns).

Note: This is available only from a detail trace that includes both SQL and I/O events.

```
BMC Software -------------- I/O Analysis-Plan/PKG   -----------PERFORMANCE MGMT
SERV ==> DBIOK           INPUT    12:00:35  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=PL,TOTAL                          LINE    1 OF    3 SCROLL=> CSR
EXPAND:  SQL, LINESEL(DBIO), CATALOG, HISTORY
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    DB2 APPLICATION TRACE
         PACKAGE/          I/O   I/O   MAX    AVG
PLAN     PROGRAM          COUNT    %  IOWAIT IOWAIT
-------- --------      -------- ----- --ms-- --ms-- 0 ...20...40...60...80
RXDB2    RXSEL2M             30  93.8  21464   1505 | ******************** |
RXDB2    RXSEL4M              1   3.1     49     49 | ************         |
RXDB2    RXSEL9M              1   3.1     49     49 | ************         |
```

Select Code:

DBIOK

Parameter:

The following parameters can be specified:

trace id

Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC

Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=PL|IC|I%|MI|AI

When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by plan and package/program name (default) and ,SORT=PL is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters

"Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Keys:

PLAN is the first column and PACKAGE/PROGRAM the second column of this display and are the two summary keys used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by plan and package/program name. The display is a scrollable list of summary lines for each unique combination of plan and package/program name found in the traced I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

Expand:

Can also EXPAND to the following display (in addition to those described in "Using EXPAND" on page 385):

SQL

Display of I/O events summarized and sorted by plan and package/program name and SQL statement number (DBIOS)

DBIOS—I/O Analysis by SQL Statement (SQL STMT)

The I/O Analysis by SQL STMT panel is a scrollable list of I/O events summarized and sorted by plan and package/program name and SQL statement number (PLAN and PACKAGE/PROGRAM and SQL STMT columns).

Note: This is available only from a detail trace that includes both SQL and I/O events.

```
BMC Software -------------- I/O Analysis-SQL Stmt   -----------PERFORMANCE MGMT
SERV ==> DBIOS           INPUT    12:01:32  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=PL,TOTAL                          LINE    1 OF    3 SCROLL=> CSR
EXPAND:  LINESEL(DBIO), CATALOG, HISTORY
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    DB2 APPLICATION TRACE
         PACKAGE/ SQL      I/O   I/O   MAX    AVG
PLAN     PROGRAM  STMT    COUNT    %  IOWAIT IOWAIT
-------- -------- ---- -------- ----- --ms-- --ms-- 0 ...20...40...60...80
RXDB2    RXSEL2M   228       30  93.8  21464   1505 | ******************** |
RXDB2    RXSEL4M   115        1   3.1     49     49 | ************         |
RXDB2    RXSEL9M   228        1   3.1     49     49 | ************         |
```

Select Code:

DBIOS

Parameter:

The following parameters can be specified:

trace id

Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC

Displays total I/O events, or only synchronous or asynchronous I/Os.

,SORT=PL|IC|I%|MI|AI

When the list of summarized I/O events is displayed initially, the list is sorted alphabetically by plan and package/program name and SQL statement number (default) and ,SORT=PL is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386.

data selection parameters

"Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Keys:

PLAN is the first column, PACKAGE/PROGRAM the second column, and SQL STMT the third column of this display. They are the three summary keys used for the default sort of summarized I/O events.

Description:

Summarizes the I/O events by plan and package/program name and SQL statement number. The display is a scrollable list of summary lines for each unique combination of plan and package/program name and SQL statement number found in the traced I/O events.

Note: Display lines with no package name or SQL statement number show non-SQL-related I/O for that plan, such as catalog or directory accesses.

(See "DBIOx—I/O Analysis (Trace)" on page 383.)

DBIOT—I/O Analysis by Time (INTERVAL START)

The I/O Analysis by Time panel is a scrollable list of I/O events summarized and sorted by time (INTERVAL START column).

```
BMC Software --------------   I/O Analysis-Time     -----------PERFORMANCE MGMT
SERV ==> DBIOT           INPUT    11:30:20  INTVL=> 3  LOG=> N  TGT==> DB2F
PARM ==> ,SO=IS,TOTAL,I=10M                    LINE    1 OF    1 SCROLL=> CSR
EXPAND:  LINESEL(DBIO), CATALOG
OPTION:  TOTAL, SYNC, ASYNC

19MAR01                    CIR2.DB2F.JQSIO.MAR19.T1818.V01
INTERVAL                   I/O   I/O   MAX    AVG
START                     COUNT    %  IOWAIT IOWAIT
--------               -------- ----- --ms-- --ms-- 0 ...20...40...60...80
18.20.00                     71    70   1028     51 | ************         |
18.30.00                     29    30    640     65 | ****************     |
```

Select Code:

DBIOT

Parameter:

The following parameters can be specified:

trace id
: Unique ID identifying the trace.

,TOTAL|SYNC|ASYNC
: Displays total I/O events, or only synchronous or asynchronous I/Os.

,INTVL=nnM|nnH
: This parameter allows a different time interval to be selected. When the list of summarized I/O events is displayed initially, the list is summarized by hour (default) and ,INTVL=1H is displayed in the PARM field (the comma is a required positional parameter for INTVL).

  The summary time interval can be changed by typing nnM or nnH where

  nnM
  : Specifies minutes

  nnH
  : Specifies hours

  The first INTERVAL START displayed is determined by rounding the earliest end time of an I/O event back to an even interval start time.

  - Any minute interval that divides evenly into 60 starts at the latest prior period from an even hour. Those not evenly divisible start at the current minute.
  - Any hour interval that divides evenly into 24 starts at the latest prior period from midnight. Those not evenly divisible start at the current hour.

For example:

– If an interval is 15 minutes and the earliest end time is 12:41:30, the first interval start time would be 12:30:00.

– If an interval is 25 minutes and the earliest end time is 12:41:30, the first interval start time would be 12:41:00.

– If an interval is 4 hours and the earliest end time is 9:21:35, the first interval start time would be 08:00:00.

,SORT=IS|IC|I%|MI|AI|ISD

When the list of summarized I/O events is displayed initially, the list is sorted in ascending sequence by interval start (default) and ,SORT=IS is displayed in the PARM field. SORT specifies the first two characters of a display column as described in "Sorting the Display" on page 386. ISD (Interval Start Descending) displays the most current measurements at the top.

data selection parameters

"Selecting a Subset of I/O Events" on page 386 provides more information about the following parameters that applies to all the I/O Analysis services:

- ,DBTS=(dbname,tsname)
- ,A|AUTHID=xxxxxxxx
- ,BP=BPnn
- ,C|CONNECT=xxxxxxxx
- ,L|LOC=xxxxxxxx
- ,P|PLAN=xxxxxxxx
- ,T|TIME=hhmm[-hhmm]
- ,D|DAY=nn[-nn]

Note: Generic qualifiers can be used to specify database and table space names for the ,DBTS parameter. See "Qualifying Requests" in *Using MAINVIEW.*

Summary Key:

INTERVAL START is the first column of this display and is the summary key used for the default sort of summarized I/O events.

If there is data for only one day in the trace buffer, the format of the timestamp is hh:mm:ss. If there is data for more than one day, the format is dd-hh:mm, where dd is the relative day number from the date of the first I/O event record in the trace buffer.

Each I/O event record traced is assigned to a time period based on its end timestamp. Therefore, the processing done for that unit of work is assigned to one interval, but work may have partially occurred during one or more previous time intervals depending on its elapsed time.

Description:

Summarizes the I/O events by time interval start. The display is a scrollable list of summary lines by a start timestamp of each summarized time interval found in the trace I/O events. (See "DBIOx—I/O Analysis (Trace)" on page 383.)

Part 3. Appendixes/Glossary

The appendixes list the IFCIDs used by ATRAC data collection and lock types for detail trace events.

Following the appendixes is a glossary of terms for all MAINVIEW products.

Appendix A. Lock Type Table 405

Appendix B. IFCID Table (ATRAC Data Collection) 409
TYPE=D Data Collection. 410
TYPE=SQL Data Collection 412
TYPE=SCAN Data Collection. 413
TYPE=IO Data Collection 413
TYPE=LOCK Data Collection. 414
TYPE=DDF Data Collection 414
TYPE=DDFVTAM Data Collection 415

Glossary 417

Appendix A. Lock Type Table

Table 10 defines the possible lock types and resources for the lock-related detail trace events:

- Detail logical lock - IFCID 21
- Lock Suspension - IFCID 44/45
- Deadlock - IFCID 172
- Timeout - IFCID 196
- Detail physical locks (P-Locks) - IFCIDs 251, 259

Table 10. Lock Types

Lock Type (short)	Lock Type (long)	Resource Name	Resource Number (hex)
ALTER BP	ALTER BUFFERPOOL	BP = bpid	
BINDLOCK	AUTOBIND/REMOTE BIND	BINDLOCK	
CATM CAT	CATMAINT CONVERT CATALOG	DB = database OB = pageset	
CATM DIR	CATMAINT CONVERT DIRECTORY	DB = database OB = pageset	
CATM MIG	CATMAINT MIGRATION	DB = database OB = pageset	
CDB PLOK	CDB P-LOCK (DDF)	DB = database	
COLLECTN	COLLECTION	Collection ID	
CS-DRAIN	CURSOR STABILITY DRAIN	DB = database OB = pageset	Part# / 0
DATABASE	DATABASE	DB = database	
DATAPAGE	DATA PAGE	DB = database OB = pageset	Page#
DBALLOC	DBALLOC - START/STOP	DB = database OB = pageset	Part# / 0
DBCMDSER	DB CMD SERIAL	DB = database	
DBD PLOK	DBD P-LOCK	DB = database	
GBP CAST	GROUP BP CASTOUT P-LOCK	BP = bpid	
GP EX UP	GROUP DB EXCEPTION UPDATE	DX = GDBET	
GROUP BP	GROUP BP START/STOP	BP = bpid	
HASH-ANC	HASH ANCHOR	DB = database OB = pageset	Page# Anchor

Table 10. Lock Types (Continued)

Lock Type (short)	Lock Type (long)	Resource Name	Resource Number (hex)
INDEXPAG	INDEX PAGE (# = x‘000002FF’, index root page)	DB = database OB = pageset	Page# Subpage#
IX EOF	INDEX END-OF-FILE	DB = database OB = pageset	Part# / 0
IXTREEPL	INDEX MANAGER TREE P-LOCK	BP = bpid DB = database OB = pageset	
LPL/GREC	LPL/GRECP DB EXCEPTION	DB = database OB = pageset	Part# / 0
MASSDEL	MASS DELETE	DB = database OB = pageset	
OPEN	OPEN PAGESET / DATASET	DB = database OB = pageset	
PAGE PLK	PAGE P-LOCK	BP = bpid DB = database OB = pageset	Part# / 0 Page#
PAGESET	PAGE SET	DB = database OB = pageset	
PART-DS	PARTITIONED DATASET	DB = database OB = pageset	Part#
PARTLOCK	PARTITION LOCKING	DB = database OB = pageset	Part#
P/P CAST	PAGESET/PARTITION CASTOUT PLOK	BP = bpid DB = database OB = pageset	Part# / 0
P/P PLOK	PAGESET/PARTITION P-LOCK	BP = bpid DB = database OB = pageset	Part# / 0
REPR DBD	REPAIR DBD TEST/DIAGNOSE	DB = database OB = pageset	
RLF PLOK	RLF P-LOCK	DB = database OB = pageset	
ROW	ROW	DB = database OB = pageset	Page# RID
RR-DRAIN	REPEATABLE READ DRAIN	DB = database OB = pageset	Part# / 0
SCA ACC	SCA ACCESS - RESTART/REDO	SC = BMC-RSTP	
SERVICE	SERVICEABILITY	(not identified)	

Table 10. Lock Types (Continued)

Lock Type (short)	Lock Type (long)	Resource Name	Resource Number (hex)
SKCT	SKELETON CURSOR TABLE	PL = planname	
SKPT	SKELETON PACKAGE TABLE	PK = Collection, package, token	
SYSLGRNG	SYSLGRNG RECORDING	DB = database OB = pageset	
TABLE	TABLE	DB = database OB = pageset	
UTIL EXC	UTILITY EXCLUSIVE EXECUTION	UTEXEC	
UTILSER	UTILITY SERIALIZATION	UTSERIAL	
UTILUID	UTILITY UID	UI = utility ID	
WR-DRAIN	WRITE DRAIN	DB = database OB = pageset	Part# / 0

- Page# is 0 for unlocks.
- P-Locks do not cause timeouts or deadlocks, so do not appear for these events.

Appendix B. IFCID Table (ATRAC Data Collection)

The data collection category used with the TYPE parameter of the ATRAC trace data collection service specifies the type of data to be collected. Each type specifies a set of IFCIDs as shown in the table that follows. The BBI-SS PAS issues an internal -STA TRACE command to collect these IFCIDs.

- D (DETAIL) collects the majority of the IFCIDs required by the DTRAC service.
- SQL includes all detail events plus SQL statements.
- SCAN includes all detail events plus scan and SQL events.
- IO includes all detail events plus I/O events.
- LOCK includes all detail events plus lock requests.
- DDF includes all detail events plus all DDF events except VTAM-related DDF events.
- DDFVTAM includes all detail events plus VTAM-related DDF events.

Note: Each additional category incurs more overhead and requires larger internal trace buffers.

IFI data collection in the BBI-SS PAS pairs start and end IFCID records for an event where they exist so that the elapsed time can be calculated, and the DTRAC service shows a single event line for start/end. Some events do not have an end event; therefore, the elapsed time cannot be calculated.

TYPE=D Data Collection

D (DETAIL) starts a trace for the following IFCIDs. The display level column contains the value that causes the event to be displayed by the DTRAC service (see "Event Data" on page 259).

Table 11. TYPE=D (DETAIL) IFCID Data Collection

IFCID	DTRAC Display Level	Description
020	2	Lock summary (COMMIT/ABORT)
023	1	Utility start
024	1	Utility object/phase change
025	1	End 023
029	2	Begin EDM request
030	2	End 029
031	1	EDM full
044	3	Lock suspend or identify to IRLM
045	3	Lock resume
051	3	Shared latch resume
052	3	Shared latch wait
054	1	Lock contention due to timeout/deadlock
056	3	Exclusive latch wait
057	3	Exclusive latch resume
068	1	Begin ABORT
069	2	End 068
070	2	Begin Phase 2 commit
071	2	End 070
072	2	Begin CREATE THREAD
073	2	End 072
074	2	Begin TERM THREAD
084	2	Begin Phase 1 commit (IMS/CICS)
085	2	End 084
088	2	Begin SYNC
089	2	End 088

Table 11. TYPE=D (DETAIL) IFCID Data Collection (Continued)

IFCID	DTRAC Display Level	Description
095	2	SORT start
096	2	End 095
107	2	OPEN/CLOSE TS
108	1	BIND/REBIND
109	1	End 108
110	1	BIND FREE
111	1	End 110
112	2	Plan allocated
125	2	Multi-index access path
140	1	Authorization failure
167	2	Conversation request queued
168	2	Distributed SQL statement
169	2	Authorization ID translation either OUTBOUND or INBOUND
170	3	Begin WAIT-SERVER
171	3	End 170
172	1	Deadlock details
177	2	Package allocation
190	2	Hybrid join failure
196	1	Lock timeout details
213	3	Begin wait for DRAIN request
214	3	End 213
215	3	Begin wait for pending CLAIMs to go to zero
216	3	End 215
218	2	Summary of COMMIT log sequence numbers
224	1	Invalid select procedure
226	1	Page latch wait
227	1	End 226
237	1	Set current degree

TYPE=SQL Data Collection

SQL collects the following IFCIDs plus those for a detail trace shown in the preceding table, which are displayed by the DTRAC service (see "Event Data" on page 259).

Table 12. TYPE=SQL IFCID Data Collection

IFCID	DTRAC Display Level	Description
022	2	Record the generated mini-plans
053	1	End DESCRIBE, COMMIT, ROLLBACK, or SQL statement error
055	1	SET SQL ID
058	1	End 059, 060, 061, 062, 063, 064, 065, or 066
059	1	SQL FETCH
060	1	SQL SELECT
061	1	SQL INSERT/UPDATE/DELETE
062	1	SQL DDL
063	2	BIND (TEXT) (unpaired)
064	1	SQL PREPARE
065	1	SQL OPEN CURSOR
066	1	SQL CLOSE CURSOR
233	2	Stored procedure CALL statement
272	2	ASSOCIATE LOCATORS statement (DB2 version 5 and later)
273	2	ALLOCATE CURSOR statement (DB2 version 5 and later)
311	2	Temporary table (DB2 version 5 and later)
324	2	User-defined function resolution (DB2 version 6 and later)
325	2	Start and end of trigger activation (DB2 version 6 and later)

TYPE=SCAN Data Collection

SCAN collects the following IFCIDs plus those for an SQL and detail trace shown in the preceding tables, which are displayed by the DTRAC service (see "Event Data" on page 259).

Table 13. TYPE=SCAN IFCID Data Collection

IFCID	DTRAC Display Level	Description
013	3	Begin hash scan
014	3	End 013
015	3	Begin index scan
016	3	Begin insert
017	3	Begin sequential scan
018	3	End 015, 016, or 017
221	3	Degree of parallel I/O processing for a parallel group
222	3	Elapsed time of a parallel group
305	2	Table check constraint

TYPE=IO Data Collection

IO collects the following IFCIDs plus those for a detail trace shown in Table 11 on page 410, which are displayed by the DTRAC service (see "Event Data" on page 259).

Table 14. TYPE=IO IFCID Data Collection

IFCID	DTRAC Display Level	Description
006	3	Begin READ I/O
007	3	End 006
008	3	Begin SYNC WRITE I/O
009	3	End 008
127	3	Begin WAIT-IO
128	3	End 127

TYPE=LOCK Data Collection

LOCK collects the following IFCIDs plus those for a detail trace shown in Table 11 on page 410, which are displayed by the DTRAC service (see "Event Data" on page 259).

Table 15. TYPE=LOCK IFCID Data Collection

IFCID	DTRAC Display Level	Description
021	3	Lock request
211	3	Claim request
212	3	Drain request
223	3	Detail of successful COMMIT LSNs (lock avoidance)
251	3	P-lock operation
259	3	Page P-lock request

TYPE=DDF Data Collection

DDF collects the following IFCIDs plus those for a detail trace shown in Table 11 on page 410, which are displayed by the DTRAC service (see "Event Data" on page 259).

Table 16. TYPE=DDF IFCID Data Collection

IFCID	DTRAC Display Level	Description
157	2	DRDS request at requesting site
158	2	RDS request at serving site
159	2	Conversation request
160	3	VTAM request at requesting site
161	3	VTAM request at serving site
162	2	Distribution Transaction Manager request at requesting site
163	2	Distribution Transaction Manager request at serving site
183	2	Requesting agent data

TYPE=DDFVTAM Data Collection

DDFVTAM collects the following IFCIDs plus those for a detail trace shown in Table 11 on page 410, which are displayed by the DTRAC service (see "Event Data" on page 259).

Table 17. TYPE=DDFVTAM IFCID Data Collection

IFCID	DTRAC Display Level	Description
164	3	VTAM exit entered
165	3	VTAM macro issued
166	3	Buffer send or receive

Glossary

This glossary defines BMC Software terminology. Other dictionaries and glossaries may be used in conjunction with this glossary.

Since this glossary pertains to BMC Software-related products, some of the terms defined may not appear in this book.

To help you find the information you need, this glossary uses the following cross-references:

Contrast with	Indicates a term that has a contrary or contradictory meaning.
See	Indicates an entry that is a synonym or contains expanded information.
See also	Indicates an entry that contains related information.

A

action. Defined operation, such as modifying a MAINVIEW window, that is performed in response to a command. *See* object.

active window. Any MAINVIEW window in which data can be refreshed. *See* alternate window, current window, window.

administrative view. Display from which a product's management tasks are performed, such as the DSLIST view for managing historical data sets. *See* view.

ALT WIN field. Input field that allows you to specify the window identifier for an alternate window where the results of a hyperlink are displayed. *See* alternate window.

Alternate Access. *See* MAINVIEW Alternate Access.

alternate form. View requested through the FORM command that changes the format of a previously displayed view to show related information. *See also* form, query.

alternate window. (1) Window that is specifically selected to display the results of a hyperlink. (2) Window whose identifier is defined to the ALT WIN field. *Contrast with* current window. *See* active window, window, ALT WIN field.

analyzer. (1) Online display that presents a snapshot of status and activity data and indicates problem areas. (2) Component of CMF MONITOR. *See* CMF MONITOR Analyzer.

application. (1) Program that performs a specific set of tasks within a MAINVIEW product. (2) In MAINVIEW VistaPoint, combination of workloads to enable display of their transaction performance data in a single view.

application trace. *See* trace.

ASCH workload. Workload comprising Advanced Program-to-Program Communication (APPC) address spaces.

AutoCustomization. Online facility for customizing the installation of products. AutoCustomization provides an ISPF panel interface that both presents customization steps in sequence and provides current status information about the progress of the installation.

automatic screen update. Usage mode wherein the currently displayed screen is refreshed automatically with new data at an interval you specify. Invoked by the ASU command.

B

batch workload. Workload consisting of address spaces running batch jobs.

BBI. Basic architecture that distributes work between workstations and multiple OS/390 targets for BMC Software MAINVIEW products.

BBI-SS PAS. *See* BBI subsystem product address space.

BBI subsystem product address space (BBI-SS PAS). OS/390 subsystem address space that manages communication between local and remote systems and that contains one or more of the following products:

- Command MQ for S/390
- MAINVIEW AutoOPERATOR
- MAINVIEW for CICS
- MAINVIEW for DB2
- MAINVIEW for DBCTL
- MAINVIEW for IMS Online
- MAINVIEW for MQSeries
- MAINVIEW VistaPoint (for CICS, DB2, DBCTL, and IMS workloads)

BBPARM. *See* parameter library.

BBPROC. *See* procedure library.

BBPROF. *See* profile library.

BBSAMP. *See* sample library.

BBV. *See* MAINVIEW Alternate Access.

BBXS. BMC Software Subsystem Services. Common set of service routines loaded into common storage and used by several BMC Software MAINVIEW products.

border. Visual indication of the boundaries of a window.

bottleneck analysis. Process of determining which resources have insufficient capacity to provide acceptable service levels and that therefore can cause performance problems.

C

CA-Disk. Data management system by Computer Associates that replaced the DMS product.

CAS. Coordinating address space. One of the address spaces used by the MAINVIEW windows environment architecture. The CAS supplies common services and enables communication between linked systems. Each OS/390 image requires a separate CAS. Cross-system communication is established through the CAS using VTAM and XCF communication links.

CFMON. *See* coupling facility monitoring.

chart. Display format for graphical data. *See also* graph.

CICSplex. User-defined set of one or more CICS systems that are controlled and managed as a single functional entity.

CMF MONITOR. Comprehensive Management Facility MONITOR. Product that measures and reports on all critical system resources, such as CPU, channel, and device usage; memory, paging, and swapping activity; and workload performance.

CMF MONITOR Analyzer. Batch component of CMF MONITOR that reads the SMF user and 70 series records created by the CMF MONITOR Extractor and/or the RMF Extractor and formats them into printed system performance reports.

CMF MONITOR Extractor. Component of CMF that collects performance statistics for CMF MONITOR Analyzer, CMF MONITOR Online, MAINVIEW for OS/390, and RMF postprocessor. *See* CMF MONITOR Analyzer, CMF MONITOR Online, MAINVIEW for OS/390.

CMF MONITOR Online. Component of CMF that uses the MAINVIEW window interface to present data on all address spaces, their use of various system resources, and the delays that each address space incurs while waiting for access to these resources. *See* CMF MONITOR, MAINVIEW for OS/390.

CMF Type 79 API. Application programming interface, provided by CMF, that provides access to MAINVIEW SMF-type 79 records.

CMFMON. Component of CMF MONITOR that simplifies online retrieval of information about system hardware and application performance and creates MAINVIEW SMF-type 79 records.

The CMFMON *online facility* can be used to view data in one or more formatted screens.

The CMFMON *write facility* can be used to write collected data as MAINVIEW SMF-type 79 records to an SMF or sequential data set.

CMRDETL. MAINVIEW for CICS data set that stores detail transaction records (type 6E) and abend records (type 6D). Detail records are logged for each successful transaction. Abend records are written when an abend occurs. Both records have the same format when stored on CMRDETL.

CMRSTAT. MAINVIEW for CICS data set that stores both CICS operational statistic records, at 5-minute intervals, and other records, at intervals defined by parameters specified during customization (using CMRSOPT).

column. Vertical component of a view or display, typically containing fields of the same type of information, that varies by the objects associated in each row.

collection interval. Length of time data is collected. *See also* delta mode, total mode.

command delimiter. Special character, usually a **;** (semicolon), used to stack commands typed concurrently on the COMMAND line for sequential execution.

COMMAND line. Line in the control area of the display screen where primary commands can be typed. *Contrast with* line command column.

Command MQ Automation D/S. Command MQ agents, which provide local proactive monitoring for both MQSeries and MSMQ (Microsoft message queue manager). The Command MQ agents operate at the local node level where they continue to perform functions regardless of the availability of the MQM (message queue manager) network. Functionality includes automatic monitoring and restarts of channels, queue managers, queues and command servers. In cases where automated recovery is not possible, the agents transport critical alert information to a central console.

Command MQ Automation S/390. Command MQ component, which monitors the MQM (message queue manager) networks and intercedes to perform corrective actions when problems arise. Solutions include:

- Dead-Letter Queue management
- System Queue Archival
- Service Interval Performance solutions
- Channel Availability

These solutions help ensure immediate relief to some of the most pressing MQM operations and performance problems.

Command MQ for D/S. Command MQ for D/S utilizes a true client/server architecture and employs resident agents to provide configuration, administration, performance monitoring and operations management for the MQM (message queue manager) network.

Command MQ for S/390. *See* MAINVIEW for MQSeries.

COMMON STORAGE MONITOR. Component of MAINVIEW for OS/390 that monitors usage and reconfigures OS/390 common storage blocks.

composite workload. Workload made up of a WLM workload or other workloads, which are called *constituent workloads.*

constituent workload. Member of a composite workload. Constituent workloads in a composite usually belong to a single workload class, but sometimes are mixed.

contention. Occurs when there are more requests for service than there are servers available.

context. In a Plex Manager view, field that contains the name of a target or group of targets specified with the CONTEXT command. *See* scope, service point, SSI context, target context.

CONTEXT command. Specifies either a MAINVIEW product and a specific target for that product (*see* target context) or a MAINVIEW product and a name representing one or more targets (see *SSI context*) for that product.

control statement. (1) Statement that interrupts a sequence of instructions and transfers control to another part of the program. (2) Statement that names samplers and other parameters that configure the MAINVIEW components to perform specified functions. (3) In CMF MONITOR, statement in a parameter library member used to identify a sampler in the extractor or a report in the analyzer, or to describe either component's processing requirements to the operating system.

coupling facility monitoring (CFMON). Coupling facility views that monitor the activity of your system's coupling facilities.

CPO. Customized Product Offering. Delivery and installation technique that allows any combination of BMC Software SMP/E-maintainable products to be distributed on a product tape to a customer and installed quickly. The CPO product tape contains libraries required for product customization and execution, plus SMP distribution libraries and data sets needed for application of SMP maintenance.

current data. Data that reflects the system in its current state. The two types of current data are realtime data and interval data. *Contrast with* historical data. *See also* interval data and realtime data.

current window. In the MAINVIEW window environment, window where the main dialog with the application takes place. The current window is used as the default window destination for commands issued on the COMMAND line when no window number is specified. *Contrast with* alternate window. *See* active window, window.

D

DASD. Direct Access Storage Device. (1) A device with rotating recording surfaces that provides immediate access to stored data. (2) Any device that responds to a DASD program.

DASD ADVISOR. An interactive software tool that diagnoses DASD performance problems and makes recommendations to reduce overall service time. This tool measures and reports on the operational performance of IBM and IBM-compatible devices.

data collector. Program that belongs to a MAINVIEW product and that collects data from various sources and stores the data in records used by views. For example, MAINVIEW for OS/390 data collectors obtain data from OS/390 services, OS/390 control blocks, CMF MONITOR Extractor control blocks, and other sources. *Contrast with* extractor.

delta mode. (1) In MAINVIEW for DB2 analyzer displays, difference between the value sampled at the start of the current statistics interval and the value sampled by the current analyzer request. See also *statistics interval.* (2) In CMFMON, usage mode wherein certain columns of data reflect the difference in values between one sample cycle and the next. Invoked by the DELta ON command. *See also* collection interval, sample cycle, total mode.

DFSMS. (Data Facility Storage Management System) Data management, backup, and HSM software from IBM for OS/390 mainframes.

DMR. *See* MAINVIEW for DB2.

DMS. (Data Management System) *See* CA-Disk.

DMS2HSM. Component of MAINVIEW SRM that facilitates the conversion of CA-Disk, formerly known as DMS, to HSM.

DSO. Data Set Optimizer. CMF MONITOR Extractor component that uses CMF MONITOR Extractor data to produce reports specifying the optimal ordering of data sets on moveable head devices.

E

EasyHSM. Component of MAINVIEW SRM that provides online monitoring and reporting to help storage managers use DFHSM efficiently.

EasyPOOL. Component of MAINVIEW SRM that provides control over data set allocation and enforcement of allocation and naming standards. EasyPOOL functions operate at the operating system level to intercept normal job processing, thus providing services without any JCL changes.

EasySMS. Component of MAINVIEW SRM that provides tools that aid in the conversion of DFSMS and provides enhancements to the DFSMS environment after implementation. EasySMS consists of the EasyACS functions, the SMSACSTE function, and the Monitoring and Positioning Facility.

element. (1) Data component of a data collector record, shown in a view as a field. (2) Internal value of a field in a view, used in product functions.

element help. Online help for a field in a view. The preferred term is *field help.*

Enterprise Storage Automation. Component of MAINVIEW SRM that integrates powerful event management technology and storage monitoring technology to provide significant storage automation capabilities and solutions. Storage occurrences are defined to generate events in the form of messages that provide an early warning system for storage problems and are routed to MAINVIEW AutoOPERATOR to be viewed.

Event Collector. Component for MAINVIEW for IMS Online, MAINVIEW for IMS Offline, and MAINVIEW for DBCTL that collects data about events in the IMS environment. This data is required for Workload Monitor and optional for Workload Analyzer (except for the workload trace service). This data also is recorded as transaction records (X'FA') and program records (X'F9') on the IMS system log for later use by the MAINVIEW for IMS Offline components: Performance Reporter and Transaction Accountant.

expand. Predefined link from one display to a related display. *See also* hyperlink.

extractor. Program that collects data from various sources and keeps the data control blocks to be written as records. Extractors obtain data from services, control blocks, and other sources. *Contrast with* data collector.

extractor interval. *See* collection interval.

F

fast path. Predefined link between one screen and another. To use the fast path, place the cursor on a single value in a field and press Enter. The resulting screen displays more detailed information about the selected value. *See also* hyperlink.

field. Group of character positions within a screen or report used to type or display specific information.

field help. Online help describing the purpose or contents of a field on a screen. To display field help, place the cursor anywhere in a field and press PF1 (HELP). In some products, field help is accessible from the screen help that is displayed when you press PF1.

filter. Selection criteria used to limit the number of rows displayed in a view. Data that does not meet the selection criteria is not displayed. A filter is composed of an element, an operator, and an operand (a number or character string). Filters can be implemented in view customization, through the PARm/QPARm commands, or through the Where/QWhere commands. Filters are established against elements of data.

fixed field. Field that remains stationary at the left margin of a screen that is scrolled either right or left.

FOCAL POINT. MAINVIEW product that displays a summary of key performance indicators across systems, sites, and applications from a single terminal.

form. One of two constituent parts of a view; the other is query. A form defines how the data is presented; a query identifies the data required for the view. See also *query, view.*

full-screen mode. Display of a MAINVIEW product application or service on the entire screen. There is no window information line. *Contrast with* windows mode.

G

global command. Any MAINVIEW window interface command that can affect all windows in the window area of a MAINVIEW display.

graph. Graphical display of data that you select from a MAINVIEW window environment view. *See also* chart.

H

hilevel. For MAINVIEW products, high-level data set qualifier required by a site's naming conventions.

historical data. (1) Data that reflects the system as it existed at the end of a past recording interval or the duration of several intervals. (2) Any data stored in the historical database and retrieved using the TIME command. *Contrast with* current data, interval data and realtime data.

historical database. Collection of performance data written at the end of each installation-defined recording interval and containing up to 100 VSAM clusters. Data is extracted from the historical database with the TIME command. *See* historical data.

historical data set. In MAINVIEW products that display historical data, VSAM cluster file in which data is recorded at regular intervals.

HSM. (Hierarchical Storage Management) Automatic movement of files from hard disk to slower, less-expensive storage media. The typical hierarchy is from magnetic disk to optical disk to tape.

hyperlink. (1) Preset field in a view or an EXPAND line on a display that permits you to

- Access cursor-sensitive help
- Issue commands
- Link to another view or display

The transfer can be either within a single product or to a related display/view in a different MAINVIEW product. Generally, hyperlinked fields are highlighted. (2) Cursor-activated short path from a topic or term in online help to related information. *See also* fast path.

I

Image log. Collection of screen-display records. Image logs may be created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Image log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to

the BBI-SS PAS Image log stops when both data sets are filled and the first data set is not processed by the archive program.

The TS Image log is a single data set that wraps around when full.

IMSPlex System Manager (IPSM). MVIMS Online and MVDBC service that provides Single System Image views of resources and bottlenecks for applications across one or more IMS regions and systems.

interval data. Cumulative data collected during a collection interval. Intervals usually last from 15 to 30 minutes depending on how the recording interval is specified during product customization. *Contrast with* historical data.

Note: If change is made to the workloads, a new interval will be started.

See also current data and realtime data.

InTune. Product for improving application program performance. It monitors the program and provides information used to reduce bottlenecks and delays.

IRUF. IMS Resource Utilization File (IRUF). IRUFs can be either detailed (one event, one record) or summarized (more than one event, one record). A detailed IRUF is created by processing the IMS system log through a program called IMFLEDIT. A summarized IRUF is created by processing one or more detailed IRUFs, one or more summarized IRUFs, or a combination of both, through a sort program and the TASCOSTR program.

J

job activity view. Report about address space consumption of resources. *See* view.

journal. Special-purpose data set that stores the chronological records of operator and system actions.

Journal log. Collection of messages. Journal logs are created for both the BBI-SS PAS and the BBI terminal session (TS).

The BBI-SS PAS Journal log consists of two data sets that are used alternately: as one fills up, the other is used. Logging to the BBI-SS PAS Journal log stops when both data sets are filled and the first data set is not being processed by the archive program.

The TS Journal log is a single data set that wraps around when full.

L

line command. Command that you type in the line command column in a view or display. Line commands initiate actions that apply to the data displayed in that particular row.

line command column. Command input column on the left side of a view or display. *Contrast with* COMMAND line.

Log Edit. In the MAINVIEW for IMS Offline program named IMFLEDIT, function that extracts transaction (X‘FA’) and program (X‘F9’) records from the IMS system log. IMFLEDIT also extracts certain records that were recorded on the system log by IMS. IMFLEDIT then formats the records into a file called the IMS Resource Utilization File (IRUF).

M

MAINVIEW. BMC Software integrated systems management architecture.

MAINVIEW Alarm Manager. In conjunction with other MAINVIEW products, notifies you when an exception condition occurs. MAINVIEW Alarm Manager is capable of monitoring multiple systems simultaneously, which means that MAINVIEW Alarm Manager installed on one system keeps track of your entire SYSPLEX. You can then display a single view that show exceptions for all MAINVIEW performance monitors within your OS/390 enterprise.

MAINVIEW Alternate Access. Enables MAINVIEW products to be used without TSO by providing access through EXCP and VTAM interfaces.

MAINVIEW AutoOPERATOR. Product that uses tools, techniques, and facilities to automate routine operator tasks and provide online performance monitoring, and that achieves high availability through error minimization, improved productivity, and problem prediction and prevention.

MAINVIEW control area. In the MAINVIEW window environment, first three lines at the top of the view containing the window information line and the COMMAND, SCROLL, CURR WIN, and ALT WIN lines. The control area cannot be customized and is part of the information display. *Contrast with* MAINVIEW display area, MAINVIEW window area.

MAINVIEW display area. *See* MAINVIEW window area.

MAINVIEW Explorer. Product that provides access to MAINVIEW products from a Web browser running under Windows. MAINVIEW Explorer replaces MAINVIEW Desktop.

MAINVIEW for CICS. Product (formerly MV MANAGER for CICS) that provides realtime application performance analysis and monitoring for CICS system management.

MAINVIEW for DB2. Product (formerly MV MANAGER for DB2) that provides realtime and historical application performance analysis and monitoring for DB2 subsystem management.

MAINVIEW for DBCTL. Product (formerly MV MANAGER for DBCTL) that provides realtime application performance analysis and monitoring for DBCTL management.

MAINVIEW for IMS (MVIMS) Offline. Product with a Performance Reporter component that organizes data and prints reports used to analyze IMS performance and a Transaction Accountant component that produces cost accounting and user charge-back records and reports.

MAINVIEW for IMS (MVIMS) Online. Product that provides realtime application performance analysis and monitoring for IMS management.

MAINVIEW for IP. Product that monitors OS/390 mission-critical application performance as it relates to IP stack usage. Collected data includes: connections, response time statistics, application availability, application throughput, and IP configuration.

MAINVIEW for MQSeries. Delivers comprehensive capabilities for configuration, administration, performance monitoring and operations management for an entire MQM (message queue manager) network.

MAINVIEW for OS/390. System management application (formerly MAINVIEW **for MVS (**prior to version 2.5)). Built upon the MAINVIEW window environment architecture, it uses the window interface to provide access to system performance data and other functions necessary in the overall management of an enterprise.

MAINVIEW for UNIX System Services. System management application that allows you to monitor the performance of the Unix System Services from a MAINVIEW window interface.

MAINVIEW for VTAM. Product that displays application performance data by application, transaction ID, and LU name. This collected data includes: connections, response time statistics, application availability, and application throughput.

MAINVIEW for WebSphere. Product that provides Web monitoring and management for applications integrated with IBM's WebSphere Application Server for OS/390.

MAINVIEW Selection Menu. ISPF selection panel that provides access to all MAINVIEW windows-mode and full-screen mode products.

MAINVIEW Storage Resource Manager (SRM). Suite of products that assist in all phases of OS/390 storage management. MAINVIEW SRM consists of components that perform automation, reporting, trend analysis, and error correction for storage management in OS/390.

MAINVIEW SYSPROG Services. *See* SYSPROG Services.

MAINVIEW VistaPoint. Product that provides enterprise-wide views of performance. Application and workload views are available for CICS, DB2, DBCTL, IMS, and OS/390. Data is summarized at the level of detail needed; e.g., reports may be for a single target, an OS/390 image, or an entire enterprise.

MAINVIEW window area. Portion of the information display that is not the control area and in which views are displayed and windows opened. It includes all but the first three lines of the information display. *Contrast with* MAINVIEW control area.

monitor. Online service that measures resources or workloads at user-defined intervals and issues warnings when user-defined thresholds are exceeded.

MV MANAGER for CICS. *See* MAINVIEW for CICS.

MV MANAGER for DB2. *See* MAINVIEW for DB2.

MVALARM. *See* MAINVIEW Alarm Manager.

MVCICS. *See* MAINVIEW for CICS.

MVDB2. *See* MAINVIEW for DB2.

MVDBC. *See* MAINVIEW for DBCTL.

MVIMS. *See* MAINVIEW for IMS.

MVMQ. *See* MAINVIEW for MQSeries

MVMVS. *See* MAINVIEW for OS/390.

MVScope. MAINVIEW for OS/390 application that traces both CPU usage down to the CSECT level and I/O usage down to the channel program level.

MVSRM. *See* MAINVIEW Storage Resource Manager (SRM).

MVSRMHSM. *See* EasyHSM.

MVSRMSGC. *See* SG-Control.

MVSRMSGD. *See* StorageGUARD.

MVSRMSGP. *See* StorageGUARD.

MVVP. *See* MAINVIEW VistaPoint.

MVVTAM. *See* MAINVIEW for VTAM.

MVWEB. *See* MAINVIEW for WebSphere.

N

nested help. Multiple layers of help pop-up windows. Each successive layer is accessed by hyperlinking from the previous layer.

O

object. Anything you can manipulate as a single unit. MAINVIEW objects can be any of the following: product, secondary window, view, row, column, or field.

You can issue an action against an object by issuing a line command in the line command column to the left of the object. *See* action.

OMVS workload. Workload consisting of OS/390 OpenEdition address spaces.

online help. Help information that is accessible online.

OS/390 and z/OS Installer. BMC Software common installation system for mainframe products.

OS/390 product address space (PAS). Address space containing OS/390 data collectors, including the CMF MONITOR Extractor. Used by MAINVIEW for OS/390,

MAINVIEW for USS, and CMF MONITOR products. *See* PAS.

P

parameter library. Data set comprised of members containing parameters for specific MAINVIEW products or a support component. There can be several versions:

- The distributed parameter library, called BBPARM
- A site-specific parameter library or libraries

  These can be

  – A library created by AutoCustomization, called UBBPARM

  – A library created manually, with a unique name

PAS. Product address space. Used by the MAINVIEW products. Contains data collectors and other product functions. *See* OS/390 product address space (PAS), BBI subsystem product address space (BBI-SS PAS).

performance group workload. MVS/SP-defined collection of address spaces. *See* service class workload, workload definition.

PERFORMANCE MANAGER. MAINVIEW for CICS online service for monitoring and managing current performance of CICS regions.

Performance Reporter (MVIMS Offline). MVIMS Offline component that organizes data and prints reports that can be used to analyze IMS performance.

Performance Reporter. Product component that generates offline batch reports. The following products can generate these reports:

- MAINVIEW for DB2
- MAINVIEW for CICS

Plex Manager. Product through which cross-system communication, MAINVIEW security, and an SSI context are established and controlled. Plex Manager is shipped with MAINVIEW window environment products as part of the coordinating address space (CAS) and is accessible as a menu option from the MAINVIEW Selection Menu.

pop-up window. Window containing help information that, when active, overlays part of the window area. A pop-up panel is displayed when you issue the HELP command.

PRGP workload. In MVS/SP 5.0 or earlier, or in compatibility mode in MVS/SP 5.1 or later, composite of service classes. MAINVIEW for OS/390 creates a performance group workload for each performance group defined in the current IEAIPSxx member.

procedure library. Data set comprised of members containing executable procedures used by MAINVIEW AutoOPERATOR. These procedures are execute command lists (EXECs) that automate site functions. There can be several versions:

- The distributed parameter library, called BBPROC
- A site-specific parameter library or libraries

  These can be

  – A library created by AutoCustomization, called UBBPROC

  – A library created manually, with a unique name

The site-created EXECs can be either user-written or customized MAINVIEW AutoOPERATOR-supplied EXECs from BBPROC.

product address space. *See* PAS.

profile library. Data set comprised of members containing profile information and cycle refresh definitions for a terminal session connected to a BBI-SS PAS. Other members are dynamically created by MAINVIEW applications. There can be several versions:

- The distributed profile library, called BBPROF
- A site-specific profile library or libraries

  These can be

  – A library created by AutoCustomization, called SBBPROF

  – A library created manually, with a unique name

The site library is a common profile shared by all site users. The terminal session CLIST creates a user profile automatically if one does not exist; it is called userid.BBPROF, where userid is your logon ID. User profile libraries allow each user to specify unique PF keys, CYCLE commands, target system defaults, a Primary Option Menu, and a unique set of application profiles.

Q

query. One of two constituent parts of a view; the other is form. A query defines the data for a view; a form defines the display format. *See also* form, view.

R

realtime data. Performance data as it exists at the moment of inquiry. Realtime data is recorded during the smallest unit of time for data collection. *Contrast with* historical data. *See also* current data and interval data.

Resource Analyzer. Online realtime displays used to analyze IMS resources and determine which are affected by specific workload problems.

Resource Monitor. Online data collection services used to monitor IMS resources and issue warnings when defined utilization thresholds are exceeded.

row. (1) Horizontal component of a view or display comprising all the fields pertaining to a single device, address space, user, etc. (2) Horizontal component of a DB2 table consisting of a sequence of values, one for each column of the table.

RxD2. Product that provides access to DB2 from REXX. It provides tools to query the DB2 catalog, issue dynamic SQL, test DB2 applications, analyze EXPLAIN data, generate DDL or DB2 utility JCL, edit DB2 table spaces, perform security administration, and much more.

S

sample cycle. Time between data samples.

For the CMF MONITOR Extractor, this is the time specified in the extractor control statements (usually 1 to 5 seconds).

For realtime data, the cycle is not fixed. Data is sampled each time you press Enter.

sample library. Data set comprised of members each of which contains one of the following:

- Sample JCL that can be edited to perform specific functions
- A macro that is referenced in the assembly of user-written services
- A sample user exit routine

There can be several versions:

- The distributed sample library, called BBSAMP
- A site-specific sample library or libraries

  These can be

  – A library created by AutoCustomization, called UBBSAMP

  – A library created manually, with a unique name

sampler. Program that monitors a specific aspect of system performance. Includes utilization thresholds used by the Exception Monitor. The CMF MONITOR Extractor contains samplers.

SBBPROF. *See* profile library.

scope. Subset of an SSI context. The scope could be all the data for the context or a subset of data within the context. It is user- or site-defined. *See* SSI context, target.

screen definition. Configuration of one or more views that have been stored with the SAVEScr command and assigned a unique name. A screen includes the layout of the windows and the view, context, system, and product active in each window.

selection view. In MAINVIEW products, view displaying a list of available views.

service class workload. OS/390- or MAINVIEW for OS/390-defined collection of address spaces.

If you are running MVS Workload Manager (WLM) in goal mode, MAINVIEW for MVS creates a service class workload for each service class that you define through WLM definition dialogs.

If you are running MVS 4.3 or earlier, or MVS/SP 5.1 or later with WLM in compatibility mode, MVS creates a performance group workload instead of a service class. *See* performance group workload.

service objective. Workload performance goal, specified in terms of response time for TSO workloads or turnaround time for batch workloads. Performance group workloads can be measured by either objective. Composite workload service objectives consist of user-defined weighting factors assigned to each constituent workload. There are no OS/390-related measures of service for started task workloads.

service point. Specification, to MAINVIEW, of the services required to enable a specific product. Services may be actions, selectors, or views. Each target (e.g., CICS, DB2, or IMS,) has its own service point.

The PLEX view lists all the defined service points known to the CAS to which the terminal session is connected.

service request block (SRB). Control block that represents a routine to be dispatched. SRB mode routines generally perform work for the operating system at a high priority. An SRB is similar to a task control block (TCB) in that it identifies a unit of work to the system. *See also* task control block.

service select code. Code entered to invoke analyzers, monitors, and general services. This code is also the name of the individual service.

session. Total period of time an address space has been active. A session begins when monitoring can be performed. If the product address space (PAS) starts after the job, the session starts with the PAS.

SG-Auto. Component of MAINVIEW SRM that provides early warning notification of storage anomalies and automated responses to those anomalies based on conditions in the storage subsystem.

SG-Control. Component of MAINVIEW SRM that provides real-time monitoring, budgeting, and control of DASD space utilization.

single system image (SSI). Feature of the MAINVIEW window environment architecture that allows you to view and perform actions on multiple OS/390 systems as though they were a single system. The rows of a single tabular view can contain rows from different OS/390 images.

SRB. *See* service request block.

SSI. *See* single system image.

SSI context. Name created to represent one or more targets for a given product. *See* context, target.

started task workload. Address spaces running jobs that were initiated programmatically.

statistics interval. For MAINVIEW for DB2, cumulative count within a predefined interval (30-minute default set by the DB2STATS parameter in the distributed BBPARM member BBIISP00) for an analyzer service DELTA or RATE display. Specifying the DELTA parameter displays the current value as the difference between the value sampled by the current analyzer request and the value sampled at the start of the current interval. Specifying the RATE parameter displays the current value by minute (DELTA divided by the number of elapsed minutes).

StopX37/II. Component of MAINVIEW SRM that provides enhancements to OS/390 space management, reducing the incidence of space-related processing problems. The StopX37/II functions operate at the system level to intercept abend conditions or standards violations, thus providing services without any JCL changes.

StorageGUARD. Component of MAINVIEW SRM that monitors and reports on DASD consumption and provides historical views to help control current and future DASD usage.

summary view. View created from a tabular view using the Summarize option in view customization. A summary view compresses several rows of data into a single row based on the summarize criteria.

SYSPROG services. Component of MAINVIEW for OS/390. Over 100 services that detect, diagnose, and correct OS/390 system problems as they occur. Accessible from the OS/390 Performance and Control Main Menu. Note that this is also available as a stand-alone product MAINVIEW SYSPROG Services.

system resource. *See* object.

T

target. Entity monitored by one or more MAINVIEW products, such as an OS/390 image, IMS or DB2 subsystem, CICS region, or related workloads across systems. *See* context, scope, SSI context.

target context. Single target/product combination. *See* context.

TASCOSTR. MAINVIEW for IMS Offline program that summarizes detail and summary IMS Resource Utilization Files (IRUFs) to be used as input to the offline components.

task control block (TCB). Address space-specific control block that represents a unit of work that is dispatched in the address space in which it was created. *See also* service request block.

TCB. *See* task control block.

terminal session (TS). Single point of control for MAINVIEW products, allowing data manipulation and data display and providing other terminal user services for MAINVIEW products. The terminal session runs in a user address space (either a TSO address space or a standalone address space for EXCP/VTAM access).

TDIR. *See* trace log directory.

threshold. Specified value used to determine whether the data in a field meets specific criteria.

TLDS. *See* trace log data set.

total mode. Usage mode in CMFMON wherein certain columns of data reflect the cumulative value between collection intervals. Invoked by the DELta OFF command. *See also* collection interval, delta mode.

trace. (1) Record of a series of events chronologically listed as they occur. (2) Online data collection and display services that track transaction activity through DB2, IMS, or CICS.

trace log data set (TLDS). Single or multiple external VSAM data sets containing summary or detail trace data for later viewing or printing. The trace log(s) can be defined as needed or dynamically allocated by the BBI-SS PAS. Each trace request is assigned its own trace log data set(s).

trace log directory (TDIR). VSAM linear data set containing one entry for each trace log data set. Each entry indicates the date and time of data set creation, the current status of the data set, the trace target, and other related information.

transaction. Specific set of input data that initiates a predefined process or job.

Transaction Accountant. MVIMS Offline component that produces cost accounting and user charge-back records and reports.

TS. *See* terminal session.

TSO workload. Workload that consists of address spaces running TSO sessions.

U

UAS. *See* user address space.

UBBPARM. *See* parameter library.

UBBPROC. *See* procedure library.

UBBSAMP. *See* sample library.

user address space. Runs a MAINVIEW terminal session (TS) in TSO, VTAM, or EXCP mode.

User BBPROF. *See* profile library.

V

view. Formatted data within a MAINVIEW window, acquired from a product as a result of a view command or action. A view consists of two parts: query and form. *See also* form, job activity view, query.

view definition. Meaning of data that appears online, including source of data, selection criteria for data field inclusion and placement, data format, summarization, context, product, view name, hyperlink fields, and threshold conditions.

view command. Name of a view that you type on the COMMAND line to display that view.

view command stack. Internal stack of up to 10 queries. For each command, the stack contains the filter parameters, sort order, context, product, and timeframe that accompany the view.

view help. Online help describing the purpose of a view. To display view help, place the cursor on the view name on the window information line and press PF1 (HELP).

W

window. Area of the MAINVIEW screen in which views and resources are presented. A window has visible boundaries and can be smaller than or equal in size to the MAINVIEW window area. *See* active window, alternate window, current window, MAINVIEW window area.

window information line. Top border of a window. Shows the window identifier, the name of the view displayed in the window, the system, the scope, the product reflected by the window, and the timeframe for which the data in the window is relevant. *See also* window status field.

window number. Sequential number assigned by MAINVIEW to each window when it is opened. The window number is the second character in the window status field. *See also* window status field.

window status. One-character letter in the window status field that indicates when a window is ready to receive commands, is busy processing commands, is not to be updated, or contains no data. It also indicates when an error has occurred in a window. The window status is the first character in the window status field. *See also* window information line, window status field.

window status field. Field on the window information line that shows the current status and assigned number of the window. *See also* window number, window status.

windows mode. Display of one or more MAINVIEW product views on a screen that can be divided into a maximum of 20 windows. A window information line defines the top border of each window. *Contrast with* full-screen mode.

WLM workload. In goal mode in MVS/SP 5.1 and later, a composite of service classes. MAINVIEW for OS/390 creates a workload for each WLM workload defined in the active service policy.

workflow. Measure of system activity that indicates how efficiently system resources are serving the jobs in a workload.

workload. (1) Systematic grouping of units of work (e.g., address spaces, CICS transactions, IMS transactions) according to classification criteria established by a system administrator. (2) In OS/390, group of service classes within a service definition.

workload activity view. Tracks workload activity as the workload accesses system resources. A workload activity view measures workload activity in terms of resource consumption and how well the workload activity meets its service objectives.

Workload Analyzer. Online data collection and display services used to analyze IMS workloads and determine problem causes.

workload definition. Workload created through the WKLIST view. Contains a unique name, a description, an initial status, a current status, and selection criteria by which address spaces are selected for inclusion in the workload. *See* Workload Definition Facility.

Workload Definition Facility. In MAINVIEW for OS/390, WKLIST view and its associated dialogs through which workloads are defined and service objectives set.

workload delay view. Tracks workload performance as the workload accesses system resources. A workload delay view measures any delay a workload experiences as it contends for those resources.

Workload Monitor. Online data collection services used to monitor IMS workloads and issue warnings when defined thresholds are exceeded.

workload objectives. Performance goals for a workload, defined in WKLIST. Objectives may include measures of performance such as response times and batch turnaround times.

Part 4. Index

This part contains the index.

Index

Symbols

* (pop-up display) 259
+ (plus)
  qualifier
    starting a trace 16, 17

A

A line command (ARCHIVE) 47
accounting
  detail report
    long 75
    short 77
  level data 96
  record 93
  summary report 73
ALLOCATE CURSOR
  DTRAC pop-up 275
application trace 93–253
archiving
  log data set 48
ASSOCIATE LOCATORS
  DTRAC pop-up 276
ATRAC service 13–31, 93
AUTHID
  I/O analysis 390
  keyword 62
  trace summary 110, 172
authorization
  failure pop-up 277
  trace 8
AUTH-XLAT pop-up 278

B

batch maintenance jobs 48
batch reports
  printing a trace 51
batch trace print 51
batch trace utilities 51
bind
  pop-up 279
  text pop-up 283
BLKDMRW 169
blocking requests
  starting a trace 13
BP keyword 64
buffer
  activity 148, 220
  I/O analysis 391
  incomplete
    LTRAC display 120
    solution 87
    STRAC display 198, 207
    TSTAT display 135
  size 10
  trace 9

C

cache
  statistics
    dynamic SQL 147, 219
check constraints
  DTRAC pop-up 362
CLAIM pop-up 284
CLAIM-WAIT pop-up 286
COMMIT-LSN pop-up 287
CONNECT keyword 62
connect trace summary 174
connection
  I/O analysis 392
  name summary 110
CONNTYPE keyword 17
constraints
  DTRAC pop-up 362
CONV-QUEUED pop-up 288
CONV-REQ pop-up 289
CORR keyword 63
correlation ID
  summary 110
  trace summary 180
Current Traces application 33

D

D line command (DELETE) 47
data access paths 96
data collection
  trace 7
data entry panel
  detail trace options 21
  modify request 37
  replicate request 31
  start trace request 14
  trace exception filters 18
  trace logging options 28
data sharing
  buffer pool
    global 150, 222
  global lock activity 153, 225
  global lock contention 292
DATE keyword 63
DB/TS
  I/O analysis 388
DB2
  accounting record 93
  traces
    logging of records 28
    use 7, 95
DB2AUTH keyword 16
DB2CONN keyword 17
DB2CORR keyword 17
DB2LOC keyword 17
DB2PKG keyword
  printing a trace 62
  starting a trace 17

DB2PLAN keyword 16
DBIO display 388
DBIO report
  keyword 68, 69
  summarization 69
DBIOA display 390
DBIOC display 392
DBIOF display 391
DBIOK display 396
DBIOL display 393
DBIOP display 394
DBIOS display 398
DBIOT display 400
DBRM
  overview (STRAC) 237
  pop-up (STRAC) 239
DBTS keyword 64
DDF
  activity information 158, 233
  IFCID data collection 414
  STRAC service 233
  TSTAT service 158
DDFVTAM data collection 415
DDN keyword 61
DDNAME keyword 61
deadlock pop-up 290
defining trace log data set 48
DETAIL 96
  IFCID data collection 410
detail trace 9, 21
  display fields 258, 381
  entry display 255
  recommendations for starting 25
  storage requirements 25
DIST-SQL pop-up 293
DRAIN pop-up 294
DRAIN-WAIT pop-up 296
DTL parameter 126
DTRAC
  display 111, 255
  keyword 66
dynamic SQL
  cache counts 147, 219
DZJPTRAC skeleton JCL member 46
DZTBARC sample member 48
DZTBPRNT program 56
DZTBRLOD sample member 48
DZTBVUNL utility 53

E

E line command (RESET) 47
EDM pool
  reports
    failures sample report 81
event pop-up displays 274
Exception Filters 18, 108
exception traces 98, 108
EXEC
  starting a trace 13
expand 40, 111
EXPLAIN pop-up 298

F

F line command (FREE) 47
filters 18
FREE
  pop-up 314
function
  user-defined
    DUSER and STRAC sections 231

G

global
  keywords 59
GMWK parameter 56
graph 169
  parameter 168
  Recent Thread History option 169
  report keyword 67
  trace summary displays 168

H

HEADING keyword 71
history
  expand button 40
  trace directory 39–49
HJOIN-FAIL pop-up 315
HOT operand 15
HT application transfer 34, 39
hyperlink
  to RxD2
    from DBIOx trace displays 385, 389
    from DTRAC 299, 356
    from STRAC 246

I

I line command
  SWITCH 38
I/O analysis
  authorization ID 390
  buffer pool 391
  connection name 392
  DB/TS 388
  location name 393
  package name 396
  plan 394
  SQL statement 398
  STRAC summary 251
  summary report by DB/TS 82
  time 400
  trace 383
  TSTAT summary 162
I/O data collection 96, 413
I/O parallelism 154, 226
identification information 198
IFCID
  See Instrumentation Facility Component Identifier
IFIBUFF 84
ILOGJCL 4
incomplete trace
  LTRAC display 120
  solution 87
  STRAC display 198, 207
  TSTAT display 135
index usage 96
indicators
  key
    STRAC display 204
    TSTAT display 133
Instrumentation Facility Component Identifier 7, 8, 259, 409
  data collection 409
  DETAIL events 259, 409
  set collected 8
Instrumentation Facility Interface 7, 93
interval
  INTERVAL keyword 67
IODATA keyword 68
IOINTVL keyword 68
IOSORT keyword 68

J

JXT011 sample member 48

K

key indicators
  STRAC display 204
  TSTAT display 133
keywords
  batch reports 60

L

large traces
  log data set allocation 30, 48
  printing 56
LEVEL keyword 66
line command
  authorization 45
  history traces application 45
  MODIFY (M) line command 37
  PURGE (P) line command 38
  REPLICATE (R) line command 31, 38
  SHOW (W) request 35
  STOP (Z) line command 38
LINECNT keyword 70
LOC keyword 63
LOCATE command 43
location
  I/O analysis 393
  summary 110
  trace summary 176
lock
  analysis 27, 106
  IFCID data collection 414
  pop-up 316
    change 316
    deadlock 290
    page p-lock 323
    p-lock 335
    query 316
    summary 318
    suspension 320
    timeout 322, 366
    unlock 316
  requests 96
  sample report 81
  STRAC summary 251
  TSTAT summary 162
  types 405
  usage analysis 27, 106
log
  data sets
    allocating large 30, 48
    batch maintenance jobs 48
    for long-running threads 88
    managing 39–49
    overwriting 31
    printing 49, 51–82
    switching 29, 31, 38
  field 34
logging
  traces 28–31, 39–49
LOGTRAC keyword 15
long-running threads
  tracing 26, 88
LTRAC 109
  display 115–120
  keyword 65

M

M line command (MODIFY) 37
maintenance jobs 48
MAXPAGES keyword 71
MODIFY (M) line command 37
modifying requests
  trace logging 37
  traces 37
monitor
  request title 15
  restart 15
multiple
  index access path 347

N

N line command (NEW) 47
navigating between traces 111
NEW command 43
NEWPAGE keyword 70

O

OP(x) 8
overhead 95

P

P line command (PRINT) 45
P line command (PURGE) 38
package
  I/O analysis 396
  overview (STRAC) 237
  pop-up (STRAC) 239
page p-lock pop-up 323
parallel degree pop-up 327
parallel elapsed pop-up 330, 332
parallelism
  DUSER and STRAC sections 226
  sysplex
    DUSER and STRAC sections 227
    TSTAT section 155
  TSTAT section 154
PARM field
  current traces 33
PKG-ALLOC pop-up 325
plan
  I/O analysis 394
  summary 110
  trace summary 178
PLAN keyword 62
PLAN-ALLOC pop-up 334
p-lock pop-up 335
plus (+)
  qualifier
    starting a trace 16, 17
PMACC 45
pop-up 259
  ALLOCATE CURSOR 275
  ASSOCIATE LOCATORS 276
  asterisk (*) indicator 259
  authorization failure 277
  AUTH-XLAT 278
  BIND 279
  BIND-TEXT 283
  CLAIM 284
  CLAIM-WAIT 286
  COMMIT-LSN 287
  CONV-QUEUED 288
  CONV-REQ 289
  DEADLOCK 290
  definition 111
  DIST-SQL 293
  DRAIN 294
  DRAIN-WAIT 296
  event displays 274
  EXPLAIN 298
  FREE 314
  FRST-INSRT 348
  HJOIN-FAIL 315
  INDX-SCAN 348
  LOCK 316
  LOCK-CHG 316
  LOCK-QRY 316
  LOCK-SUMMARY 318
  LOCK-SUSP 320
  LOCK-TMO 322
  PAGE P-LOCK 323
  PKG-ALLOC 325
  PLAN-ALLOC 334
  P-LOCK 335
  PRLL-DEGREE 327
  PRLL-ELAP 330, 332
  READ-I/O 339
  REQ-AGNT 340
  REQ-DRDS 342
  REQ-DTM 343
  REQ-VTAM 344
  RIDLIST 346
  SEQ-SCAN 348
  SERV-DRDS 350
  SERV-DTM 351
  SERV-VTAM 352
  SORT 354
  SQL statement 356
  stored procedures
    entry 359
    exit 360
  SYNC-WRITE 361
  TBL-CONSTR 362
  TEMPTABLE 364
  TIMEOUT 366
  trigger
    entry 369
    exit 371
  UNLOCK 316

pop-up *(continued)*
usage 97
user-defined function 373
utility 375
VTAM-EXIT 377
VTAM-MACRO 378
WAIT-I/O 379
POPUP keyword 66
printing 49
archived TLDSs 53
traces 4, 49, 51–82
large traces 56
process table check constraint pop-up 362
PUR operand 15
PURGE (P) line command 38
purging requests
active traces 38
traces 15

Q

Q line command (QUIESCE LOGGING) 38
QIS
operand 15
qualifier
starting a trace 16, 17
quiescing logging 38

R

R line command (REPLICATE) 31, 38
read I/O pop-up 339
recommendations
detail trace 25
storage requirements 25
REPLICATE (R) line command 31, 38
report keywords 60
report samples 72
Accounting Detail — Long 75
Accounting Detail — Short 77
Accounting Summary 73
I/O Summary by DB/TS 82
Lock and EDM Pool Failures 81
SQL Statement Summary 78
SQL Text and EXPLAIN 80
Thread Detail Events 79
Workload Summary by Time 74
report selection 62
report summarization
DBIO report 69
REPORTID keyword 61
REQ-AGNT pop-up 340
REQ-DRDS pop-up 342
REQ-DTM pop-up 343
requests
modifying
traces 37
REQ-VTAM pop-up 344
resource
usage information 201
restart service 15
restoring
log data set 48
RID
number processed 347
RID list pop-up 346
routines
STRAC service 229
TSTAT service 157
RST keyword 15
RxD2
hyperlinks to
from DBIOx trace displays 385, 389
from DTRAC 299, 356
from STRAC 246

S

S line command
SELECT
current traces 35
history traces 45
sample reports 72
Accounting Detail — Long 75
Accounting Detail — Short 77
Accounting Summary 73
I/O Summary by DB/TS 82
Lock and EDM Pool Failures 81
SQL Statement Summary 78
SQL Text and EXPLAIN 80
Thread Detail Events 79
Workload Summary by Time 74
scan pop-up 348
scans 96
IFCID data collection 413
scrolling 112
buffer
increasing the size 56
SELECT (S)
list of trace entries 35
SERV-DRDS pop-up 350
SERV-DTM pop-up 351
service(s)
start time 15
stop time 15
title 15
SERV-VTAM pop-up 352
SHOW (W) line command 35
SLOGJCL 4
SORT
database lock and I/O summary 251
database summary 249
DBIO display 388
DBIOA display 390
DBIOC display 392
DBIOF display 391
DBIOK display 396

SORT *(continued)*
DBIOL display 393
DBIOP display 394
DBIOS display 398
DBIOT display 401
history traces 43
SQL summary 244
STRAC display 195
summary 253
trace summary displays 169
TSUMA display 172
TSUMC display 174
TSUML display 176
TSUMP display 178
TSUMR display 180
TSUMT display 183
sort detail pop-up 354
SORTIOLOCK keyword 66
SORTSCAN keyword 66
SORTSQL keyword 65
SORTTSUM keyword 67
SQL
activity 144, 216
dynamic
cache counts 147, 219
I/O analysis 398
IFCID data collection 96, 412
statement information 96
statement pop-up 243, 246, 356
statement summary report 78
summary 242
text and EXPLAIN report 80
ST application transfer 13, 34
START
command 44
start time 15
starting
trace 3
detail trace 21
from a BLK member 13
from an exec 13
summary trace 13
status
field 34
of requests 34
STOP
(Z) line command 38
command 44
keyword 15
stop time 15
stopping
trace 3
STORAGE keyword 14
storage requirements
detail trace 25
stored procedures
STRAC service 230
TSTAT service 157
stored procedures pop-up
entry 359
exit 360
STRAC 110
display 185–253
keyword 65
summarization
DBIO report 69
SUMMARY 96
summary trace
authorization ID 172
connection name 174
correlation ID 180
location name 176
plan name 178
starting 13
time 182
Summary Trace Entry display 185–253
switching
log data sets 29, 31, 38
sync write pop-up 361
syntax notation xviii
sysplex
parallelism
DUSER and STRAC sections 227
TSTAT section 155

T

table
check constraints
DTRAC pop-up 362
temporary
DTRAC pop-up 364
target
keyword 61
system 33
historical trace 40
tbl-constr pop-up 362
temporary table
DTRAC pop-up 364
temptable pop-up 364
THRDHIST trace ID 169
Thread Detail Events report 79
time
I/O analysis 400
summary 110
trace summary 182
TIME keyword 64
timeout pop-up 322, 366
title 15
TITLE keyword 62
display
current traces application 33
TLDS 51, 53–82

trace
application 93–253
AUTHID summary 110
authorization 8
buffers 9
current active thread 380
data collection 7
DDF 414
DDFVTAM 415
DETAIL 410
IO 413
levels 95, 259–274
LOCK 414
SCAN 413
SQL 412
detail display fields 258, 381
display
summary of services 109
display levels 111, 257, 381
options 97
displaying 35
list of entries 35, 115
single entry 185, 255
event data 259
events 111
history 39–49
I/O data collection 413
incomplete
LTRAC display 120
solution 87
STRAC display 198, 207
TSTAT display 135
level 1 display events 259
level 2 display events 259
level 3 display events 259
level selection considerations 95
list of entries 109
LOCK data collection 414
log data set 49, 51–82
logging
records 28, 39
long-running threads 26, 88
overhead 95
printing 49, 51–82
large traces 56
record summary 109, 121
SCAN data collection 413
selecting data for retention 18
starting 3, 13
detail 21
from a BLK member 13
from an exec 13
statistics 121
status 109, 121
stopping 3
storage 9
trace *(continued)*
summary
by authorization ID 172
by connection name 174
by correlation ID 180
by location 176
by plan 178
by time 182
of displays 109
usage 95
Trace Entries display 115–120
Trace Statistics display 121–164
trace summary
by authorization ID 172
by connection name 174
by correlation ID 180
by location name 176
by plan name 178
by time 182
TRACEID keyword 61
TRALLOC 30
transaction summary 110
TRARCSTC keyword 37
TRBUFF 10
Detail Trace Options panel 24
recommendations 25
TRCYL keyword 37
TRDIR keyword 39
trigger pop-up
entry 369
exit 371
triggers
DUSER and STRAC sections 232
TRNUMDS keyword 37
TRREUSE keyword 37
TRSIZE 10
Detail Trace Options panel 24
recommendations 25
TRSMSDCL keyword 37
TRSMSMCL keyword 37
TRSMSSCL keyword 37
TRSWTIME keyword 37
TRTIME parameter 5, 15
TRVOLS keyword 37
TSTAT 109
display 121–164
keyword 67
TSUM keyword 67
TSUMA 110, 172
TSUMC 110, 174
TSUML 110, 176
TSUMP 110, 178
TSUMR 110, 180
TSUMT 110, 182
TYPE
keyword 14

U

UNLOCK
    detail pop-up 316
user
    detail trace display 380–381
    ID
        field 33
user-defined function
    DTRAC pop-up 373
    DUSER and STRAC sections 231
USERID
    keyword 61
utility pop-up 375
UTRAC display 111, 380–381

V

V line command (VERIFY) 47
viewing traces 109
VTAM-EXIT pop-up 377
VTAM-MACRO pop-up 378

W

W
    line command (SHOW) 35, 45
WAIT-I/O pop-up 379
WIDTH keyword 70
Workload Summary by Time report 74
WRAP keyword 15

Z

Z line command (STOP) 38

Notes

100038546