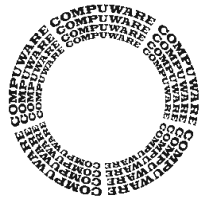
STROBE MVS

STROBE DB2 Feature

Release 3.0



COMPUWARE®

Please direct questions about STROBE MVS
or comments on this document to:

STROBE MVS Technical Support

Compuware Corporation
124 Mount Auburn Street
Cambridge MA 02138-5758
1-800-585-2802 or
1-617-661-3020
1-617-498-4010 (fax)
strobe-sup@compuware.com

Outside the USA and Canada, please contact
your local Compuware office or agent.

Doc. CWSTUD3A
January 23, 2003.

Contents

Figures

Summary of Changes

This section lists the changes from STROBE MVS for Sysplex Release 2.5 to Release 3.0.

Changes to the STROBE DB2 Feature

The STROBE DB2 Feature no longer supports DB2 Version 3 and 4.

Changes to this Manual

The technical content of this manual did not change for STROBE MVS for Sysplex Release 3.0.

Introduction

This manual describes measurement concepts applicable to and specific data made available by the STROBE DB2 Feature of the STROBE MVS Application Performance Measurement System. The STROBE DB2 Feature augments functions provided by the basic STROBE system.

The STROBE MVS Application Performance Measurement System and the STROBE DB2 Feature are products designed for IBM MVS/ESA, IBM OS/390, and IBM z/OS systems. The STROBE DB2 Feature is designed for use with IBM DATABASE 2 (DB2) Versions 5, 6 and 7 program numbers 5655-DB2, 5645-DB2, and 5675-DB2.

How This Manual Is Organized

Chapter 1, "Overview" presents an overview of the STROBE DB2 Feature and how it interacts with the DB2 operating environments.

Chapter 2, "Using the STROBE DB2 Feature" explains how to use the STROBE DB2 Feature.

Chapter 3, "The Performance Profile for a DB2 Allied Address Space" explains how to interpret the Performance Profile for a DB2 allied address space.

Chapter 4, "The STROBE Performance Profile for DB2 Services Address Spaces" explains how to interpret the Performance Profile for a DB2 database services address space (DBSAS).

How to Use This Manual

You should read Chapter 1, "Overview" and Chapter 2, "Using the STROBE DB2 Feature" before submitting a measurement request. If you want to interpret a STROBE Performance Profile for a DB2 allied address space, read Chapter 3, "The Performance Profile for a DB2 Allied Address Space". If you want to interpret a STROBE Performance Profile for a DB2 database services address space, read Chapter 4, "The STROBE Performance Profile for DB2 Services Address Spaces".

The STROBE Library

The STROBE base product manuals include:

- *STROBE MVS Concepts and Facilities*, document number CWSTGX3A

  STROBE MVS Concepts and Facilities explains how to decide which programs and online regions to measure, when to measure them, and how to interpret the reports in the STROBE Performance Profile.

- *STROBE MVS Messages,* document number CWSTXM3A

  STROBE MVS Messages lists all messages and abnormal termination (ABEND) codes, describes how to interpret them, and in many cases suggests a corrective action.

- *STROBE MVS System Programmer's Guide*, document number CWSTXI3A

  The *STROBE MVS System Programmer's Guide* explains how to install and maintain STROBE.

- *STROBE MVS User's Guide,* document number CWSTUX3A and the *STROBE MVS User's Guide with Advanced Session Management*, document number CWSTUA3A

  The *STROBE MVS User's Guide* explains how to use STROBE to measure application performance. The *STROBE MVS User's Guide with Advanced Session Management* explains how to use STROBE with the STROBE Advanced Session Management Feature to measure application performance. Users who have the STROBE Advanced Session Management Feature will use this manual rather than the *STROBE MVS User's Guide*.

- *STROBE MVS Application Performance Measurement System Quick Reference*

  The *STROBE MVS Application Performance Measurement System Quick Reference* is a convenient reference for how to use STROBE and for interpreting the STROBE Performance Profile.

STROBE Feature Manuals

These manuals describe the optional features of the STROBE MVS Application Performance Measurement System. Each manual describes measurement concepts applicable to and specific data made available by the feature.

- *STROBE MVS User's Guide with Advanced Session Management*, document number CWSTUA3A
- *STROBE ADABAS/NATURAL Feature*, document number CWSTUN3A
- *STROBE CA-IDMS Feature*, document number CWSTUR3A
- *STROBE CICS Feature*, document number CWSTUC3A
- *STROBE COOL:Gen Feature*, document number CWSTUG3A
- *STROBE CSP Feature*, document number CWSTUP3A
- *STROBE DB2 Feature,* document number CWSTUD3A
- *STROBE IMS Feature,* document number CWSTUI3A
- *STROBE Interface Feature*, document number CWSTUF3A
- *STROBE Java Feature,* document number CWSTUJ3A
- *STROBE MQSeries Feature,* document number CWSTUM3A
- *STROBE UNIX System Services Feature,* document number CWSTUU3A

Online Documentation

STROBE manuals are available in HTML, Adobe Acrobat PDF format, and IBM BookManager format, on CD-ROM and at Compuware's technical support Web site at http://frontline.compuware.com.

Online Help

STROBE products provide the following online information:

- STROBE/ISPF Online Tutorials, Option T from the STROBE/ISPF STROBE OPTIONS menu
- STROBE/ISPF Online Message Facility, Option M from the STROBE/ISPF STROBE OPTIONS menu

Other Compuware Application Performance Management Products

The following products and features work in conjunction with the STROBE MVS Application Performance Measurement System. These tools extend the benefits of application performance management (APM).

iSTROBE

iSTROBE enables you to view and analyze STROBE Performance Profile data on a workstation using a standard Web browser. Easy to install and easy to use, iSTROBE guides you through the performance analysis process and offers recommendations for improving performance. iSTROBE simplifies the performance analysis of applications that you measure with STROBE. For more information on iSTROBE, see the *iSTROBE Getting Started Guide*.

SQL Analysis Feature

The SQL Analysis Feature works in conjunction with STROBE and iSTROBE or APMpower to supply access path analyses and database and SQL coding recommendations for DB2 applications measured by STROBE. The SQL Analysis Feature pinpoints the most resource-consumptive static or dynamic SQL statements, explains why these statements might be inefficient, and provides recommendations to improve the performance of the DB2 application. For more information on the SQL Analysis Feature, see the *STROBE MVS User's Guide* or the *STROBE MVS User's Guide with Advanced Session Management*.

APMpower

The APMpower Application Performance Analysis System extends the benefits of STROBE to application developers who use workstations to develop, test, and maintain MVS applications. Developers employ the APMpower graphical user interface and advanced analytical aids to navigate the Performance Profile, analyze and improve application performance, and share performance knowledge across the IS organization. For more information about APMpower, see the APMpower documentation.

Compuware APM Technical Support

For North American customers, for technical support, please contact the Technical Support department by telephone at (800) 585-2802 or (617) 661-3020, by fax at (617) 498-4010, or by e-mail at strobe-sup@compuware.com.

To access online technical support, visit Compuware's FrontLine page on the World Wide Web at http://frontline.compuware.com and select the product "STROBE and APMpower."

For other international customers, please contact your local Compuware office or STROBE supplier.

Compuware APM Training

Compuware's Education Resources Group offers a range of training options for organizations that use STROBE, iSTROBE, and APMpower. To arrange Application Performance Management training, please contact Compuware at 1-800-835-3190 or visit Compuware's Education Resources Group at http://www.compuware.com/training

For other international customers, please contact your local Compuware office or STROBE supplier for a complete list of APM Training offerings.

Compuware APM Service Offerings

For North American customers, for information about current service offerings, please contact your local Compuware sales office or call Compuware Corporate Headquarters at 1-800-COMPUWARE (266-7892) or visit Compuware's APM Product page on the World Wide Web at http://www.compuware.com/products/strobe.

For other international customers, please contact your local Compuware office or STROBE supplier for a complete list of Services offerings.

APM Installation Assurance

The APM Installation Assurance service assists you in planning for, installing, customizing and using APM products. The service will help you maximize the value and benefits derived from the APM product family.

Consulting engineers work closely with your IT personnel to understand your operating environment and your organization's APM goals. The engineer will assist you in developing a customization and installation plan for STROBE, iSTROBE, and APMpower. The engineer will oversee the installation process and verify product readiness. The engineer will also help set up measurement request schedules, request groups, history records, AutoSTROBE measurement requests, and will verify the installation of the SQL Analysis Feature.

With APM Installation Assurance services, your organization can immediately maximize the value received from your investment in the APM product family. You will also benefit from a fully customized installation that will enhance the product functionality and increase the automation aspects of your APM initiatives.

Application Performance Management Consulting

The Application Performance Management (APM) Consulting services assist you in identifying and resolving specific performance problems in your OS/390 business-critical applications.

Using STROBE, iSTROBE, and APMpower, consulting engineers work closely with your IT personnel to measure an application's performance, identify performance improvement opportunities and make recommendations for implementing solutions.

With APM Consulting services, your organization cannot only resolve problems quickly and effectively, but also gain the skills necessary to prevent application performance degradation in the future.

Application Performance Assessment

The Application Performance Assessment (APA) service assists you in achieving a higher level of performance for your OS/390 business-critical applications.

Using STROBE, iSTROBE, and APMpower, consulting engineers work closely with your IT personnel to evaluate the efficiency of business-critical applications, identify opportunities for improving performance and document the potential savings that can result from implementing recommended solutions.

With APA services, you cannot only improve application performance quickly and effectively, but also gain the knowledge and skills necessary to implement and sustain a process-oriented application performance management (APM) program.

Chapter 1.
Overview

The STROBE MVS Application Performance Measurement System is a product that determines where and how application time is spent in online regions and batch processing programs and how system resources are used. STROBE collects several types of data as it tracks activity within an MVS environment and produces a collection of reports that helps you determine where to revise applications to improve their performance.

The STROBE DB2 Feature extends the functions of STROBE. With the STROBE DB2 Feature, you can measure and evaluate the performance of applications that use IBM DATABASE 2 (DB2). The Feature identifies programs and transactions that are most active within your DB2 system so you can detect opportunities to improve efficiency.

This chapter describes the characteristics of the STROBE DB2 Feature and its benefits.

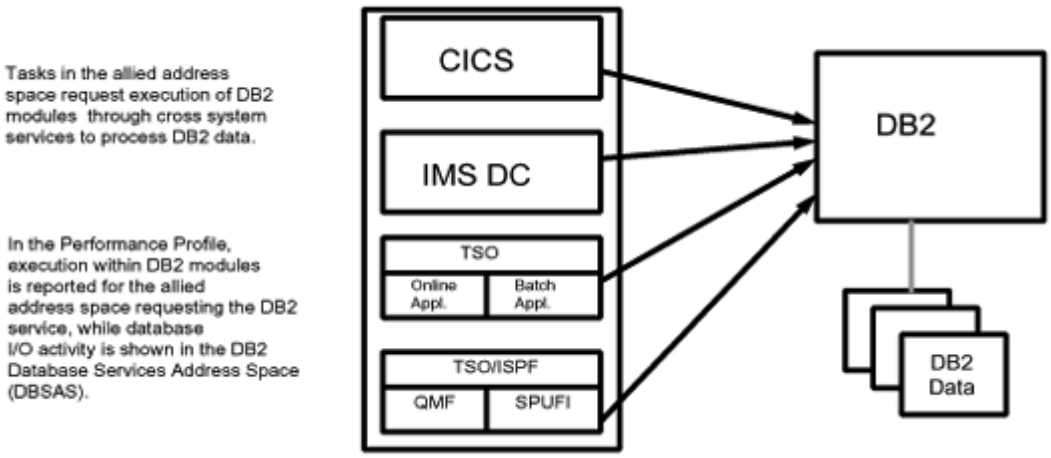
STROBE and the DB2 Operating Environments

The STROBE DB2 Feature gathers measurement data in all the environments in which DB2 functions. It gathers data for

- the allied address spaces, which consist of the batch processing programs and CICS, IMS, and TSO address spaces that use DB2 services
- the DB2 database and system services address spaces and the stored procedures address spaces

When you measure an application that uses DB2 in a CICS, IMS, or TSO environment, the resulting STROBE Performance Profile might show significant levels of activity in DB2 system modules. The STROBE DB2 Feature attributes activity in DB2 system modules to the SQL statements within the application programs that caused the activity. Figure 1-1 shows the relationship between the address spaces.

Figure 1-1. DB2 Address Spaces



DB2 Allied Address Spaces

When you measure the allied address space, STROBE monitors any application threads that make calls to DB2 database tables. If heavy CPU usage is attributed to these calls, STROBE reports this activity in program usage reports and you will see the largest CPU usage values for the DB2 system service modules.

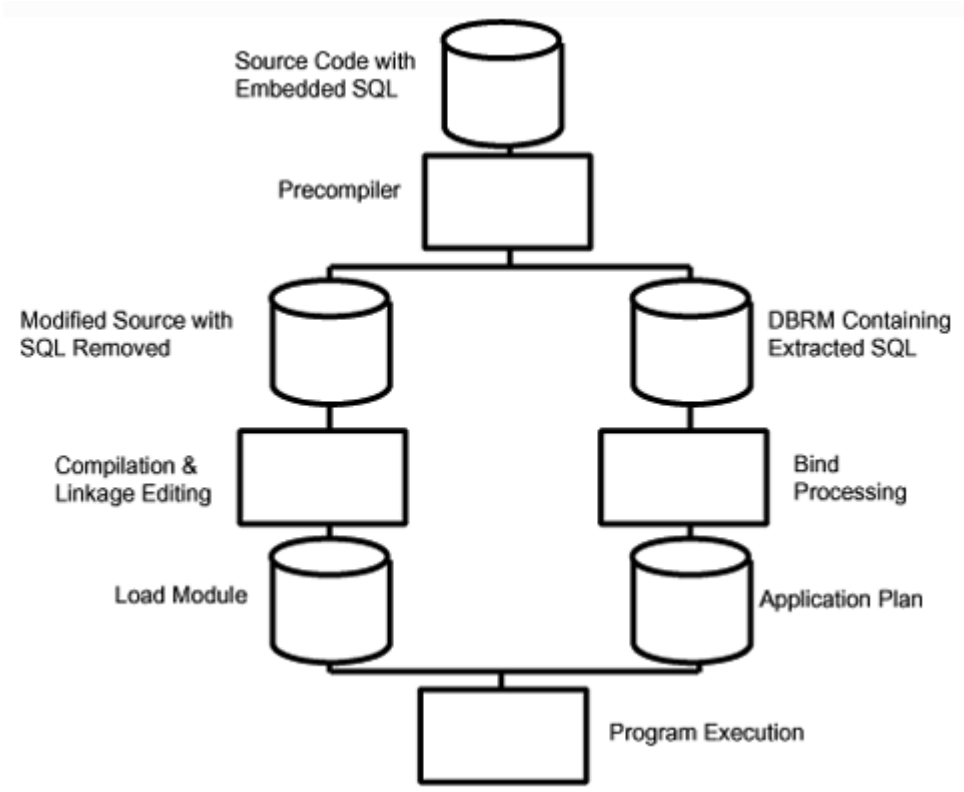
The Performance Profile for any allied address space also shows I/O activity for the data sets that have been allocated to that address space. Because DB2 databases are allocated in the DB2 database services address space (DBSAS), you must measure this address space to obtain a Performance Profile showing access to DB2 database files.

Batch jobs or online regions that access DB2 databases often generate significant amounts of CPU activity in DB2 system modules. The key to improving the performance of such applications is identifying the SQL statements that invoke DB2 services. A prime consideration is whether the SQL statements are *static* or *dynamic*.

To prepare static statements, you invoke the DB2 precompiler to process your application program. The precompiler checks the syntax of your SQL statements and replaces them with the code necessary to invoke DB2. If the syntax is correct, the precompiler copies the SQL into a database request module (DBRM), which DB2 later binds into a plan. After you issue the DB2 command to bind the plan, DB2 determines the access path. The application program can then use the plan without further binding. For an illustration of the plan preparation process, see Figure 1-2.

Dynamic SQL applications also use a DBRM, but the SQL statements in the DBRM (for example, PREPARE and EXECUTE) do not directly access data. The data-accessing SQL statements (for example, SELECT, UPDATE) are created by the application program and sent to DB2 for processing via the PREPARE and EXECUTE statements in the DBRM.

Figure 1-2. Application Program/DBRM Preparation



SQL Processor Using File Input (SPUFI) and Query Management Facility (QMF) are special types of dynamic SQL applications that create their own DBRMs to process the user-entered SQL statements.

- With SPUFI, the user can specify a series of SQL statements that are contained in a single file.
- With QMF, the user enters SQL statements via Saved Queries, Queries by Example, Prompted Queries, or Procedures. Procedures can contain Queries, calls to other Procedures (henceforth called subprocedures), and QMF commands.

For both QMF and SPUFI, the STROBE DB2 Feature differentiates between the user-entered (*target*) statements and the system-generated (*executing*) statements in the STROBE Performance Profile, and displays both types of statements in the CPU Usage by SQL Statement reports.

For QMF Procedures, the STROBE DB2 Feature also displays the structure of each QMF Procedure. The application developer must consider not only the individual SQL statements but also the relationships among SQL statements and QMF commands. STROBE describes the Procedure's structure, adding detailed reports that show the CPU usage for each Query and QMF command invoked by the Procedure.

The STROBE CPU Usage by SQL Statement reports show CPU execution as occurring in the SQL statements that are part of a Query, Procedure, or DBRM. For dynamic SQL statements, STROBE identifies both the target and the executing SQL statement. The target statements contain the most useful information for the application developer, because those are the ones that can be modified.

In some cases, information is most useful when it focuses on individual executions of Queries. When you use QMF to prototype DB2 applications, however, it may be more valuable to aggregate CPU execution from multiple Query executions into a single report. In this way you can simulate a production environment in which different users are executing the same SQL statements. STROBE offers both options for SQL reports. (For more information, see "Reporting QMF Procedure and Query Structures" on page 3-21.)

Other DB2 Information

The STROBE DB2 Feature also

- shows which DB2 service modules were active during the measurement session
- shows the SQL statement that invoked these routines
- provides function descriptors for the DB2 modules
- includes automatic control section identification for the DB2 batch processing application running under the TSO control programfor TSO address spaces and batch processing programs

DB2 Services Address Spaces

You can also monitor the performance of the following DB2 services address spaces:

- DB2 database services address space (DBSAS)
- DB2 system services address space (SSAS)
- stored procedures address space (SPAS)

However, because the DB2 system services modules are executed in cross-memory mode by the allied address spaces, most of the DB2 activity is shown in the Performance Profile for the allied address space.

Database Services Address Space

Reports in the Performance Profile for a DBSAS show

- DB2 database I/O facility usage by device, volume, data definition name, and cylinder
- data set characteristics of the databases
- CPU activity initiated by tasks in the DB2 DBSAS
- function descriptors for the DB2 modules

System Services Address Space

The Performance Profile for the system services address space shows, among others, logging and initialization functions.

Stored Procedures Address Space

The Performance Profile for the stored procedures address space shows SQL and CPU usage by stored procedures.

Benefits of the STROBE DB2 Feature

The data provided by the STROBE DB2 Feature will help you throughout the application life cycle:

- in design, to evaluate the performance of prototypes
- in development, to streamline DB2 programs so that database calls are coded efficiently
- in production and quality assurance, to ensure efficient performance of DB2 programs and services, optimizing the use of existing hardware resources
- in maintenance, to assess the impact of changes in business requirements and data volume on the performance of DB2 applications

Chapter 2.
Using the STROBE DB2 Feature

The DB2 Feature automatically starts as soon as STROBE detects DB2 activity and is initially configured to report on all types of DB2 application data. You should only need to re-configure the Feature if your Performance Profile does not contain the information you are seeking.

The following are considerations that might require you to change the DB2 Feature configuration:

- What level of data detail do you want STROBE to collect on your DB2 environment? You can specify whether you want to see the total CPU time consumed by a set of queries or procedures or examine them on an individual basis.
- Does the application you are measuring have any special characteristics? If it contains a large number of SQL statements, you can specify thresholds that statements must meet for STROBE to display their complete text in the Performance Profile.

This chapter describes how to

- collect additional SQL statement information
- limit the types of data STROBE collects
- produce and tailor STROBE Performance Profiles using STROBE and the STROBE DB2 Feature

Managing DB2 Data Collection

STROBE automatically invokes the STROBE DB2 Feature and some of its components while other components must be manually activated. This section describes how to make these specifications for STROBE. You can find detailed instructions for submitting measurement requests in the STROBE/ISPF Online Tutorials and in Chapters 2 and 3 of the *STROBE MVS User's Guide*.

Collecting Execution Count and Average Statement Information

STROBE identifies the number of times a SQL statement executes during a measurement session, as well as the average service time for each SQL statement. This information is collected by default, and is displayed in these reports

- SQL CPU Usage Summary Report
- SQL Wait Summary Report
- CPU Usage by SQL Statement Report
- Wait by SQL Statement Report

Note: The statement count and elapsed time are calculated only for SQL statements contained in DBRMs. STROBE will not make the calculations for SQL statements that are part of QMF or SPUFI queries and procedures, or for stored procedures.

Capturing Additional Execution Data

Because STROBE automatically collects execution count and service time data, you do not need to specify any additional parameters when measuring DBRMs. However, you may need to adjust the parameters STROBE uses to determine how to store statement count and service time data.

To change the parameters, complete the following steps:

1. Select Option 1 from the STROBE OPTIONS menu. STROBE/ISPF displays the STROBE - ADD ACTIVE REQUEST panel (Figure 2-1).

Figure 2-1. STROBE - ADD ACTIVE REQUEST Panel

```
------------------------- STROBE - ADD ACTIVE REQUEST -------------------------
COMMAND ===>

JOBNAME ===> IMSDC1     (Jobname or clear to list active jobs)


SYSTEM  ===> SCS01      (System or clear to list available systems)

AUTO PROFILE CREATION   ===> Y (Y or N; Use Y only when overriding defaults)

MEASUREMENT SESSION INFORMATION:
  SESSION DURATION      ===> 1        (Estimated time in minutes)
  TARGET SAMPLE SIZE    ===> 10000    (Target number of samples)

TSO USERID TO NOTIFY    ===> WPAFXC   (Notify when session completes)

SAMPLE DATA SET INFORMATION:
  DATA SET NAME PREFIX  ===> ZZ
  UNIT NAME   ===> WPAANY   VOLUME ===>        DISP ===> CATLG (CATLG OR KEEP)

SELECT ADDITIONAL PARAMETERS: (Y or N; Use Y only when overriding defaults)
  DATA COLLECTORS       ===> N     MODULE MAPPING DATA          ===> N
  SESSION MANAGEMENT    ===> N     REQUEST RETENTION            ===> N
  OTHER PARAMETERS      ===> N
```

2. Enter “Y” in the OTHER PARAMETERS field and press **Enter**. STROBE/ISPF displays the STROBE - OTHER PARAMETERS panel (Figure 2-2 on page 2-3).
3. In the OTHER PARAMETERS field, specify DB2=CAPTBUFF=*nnnn*. The value you specify is the basis used by STROBE to determine the number of kilobytes for the size of the buffer(s) in above the line storage to hold SQL data during intervals before the SQL data is removed from the buffer. If you set a low value for the target sample size, a long value for session duration and a high volume of SQL statements are executing during the measurement session, you should set CAPTBUFF to a higher value. The default value for CAPTBUFF is 100. You can specify a value anywhere from 100 to 9999. If you set CAPTBUFF to less than 100 kilobytes, the value will revert to 100.

Note: If you are running the STROBE *DB2 Feature* in a multi-thread environment, see “DB2 Multi-Thread Environments” on page 2-3 before you set values for the CAPTBUFF and MAXMEM parameters.

In the OTHER PARAMETERS field, specify DB2=MAXMEM=*nnnn*, The value you specify sets the total amount of virtual above the line storage STROBE uses for unique SQL text during the course of the measurement session. The default value for MAXMEM is 2 megabytes (DB2=MAXMEM=2000). You can specify a value anywhere from 1 to 9999. Do *not* set MAXMEM to a value less than the value specified for CAPTBUFF.

4. Press **Enter** to submit the measurement request.

Figure 2-2. STROBE - OTHER PARAMETERS Panel (Specifying Data Storage Size)

```
------------------- STROBE - OTHER PARAMETERS --------------------------------
COMMAND ===>

OTHER PARAMETERS FOR JOBNAME :   IMSTESA


OTHER PARAMETERS ===> DB2=(CAPTBUFF=100,DB2=MAXMEM=2500)
```

You can specify both a CAPTBUFF and a MAXMEM value as shown above. Do not place any spaces between any of the parameter and the values you specify for them. If you are using the STROBE command language, specify DB2=CAPTBUFF=*nnnn* or DB2=MAXMEM=*nnnn* when you submit the ADD command to define the size of the data capture buffer.

DB2 Multi-Thread Environments

If you are running the STROBE DB2 Feature with the capture option on in a multi-thread environment, above the line storage increases because multiple buffers may be created for each thread. For all environments, a default amount of above the line storage is available for the buffers. Then depending on the multi-thread environment being measured by the *DB2 Feature*, additional above the line storage is required. STROBE uses the value of the CAPTBUFF parameter to calculate the amount of this additional storage. The following lists how STROBE determines how much above the line storage to use.

All Environments

STROBE always requires 960 KB of above the line storage for buffers in a multi-thread environments. The *DB2 Feature* will retain in above the line storage the first 4 kilobytes of each unique SQL statement that it encounters during the sampling process. Either the value of MAXMEM parameter or a maximum of 2,000 kilobytes will be used.

CICS Environment

Beside the storage space required for every environment, additional buffer space is required based on the following calculation:

(CAPTBUFF value x 2) x Number of Threads

If more than 30 threads are running in the environment, STROBE requires an additional 32KB buffer for each thread over of 30.

IMS/TSO/ISPF Environments

Beside the storage space required for every environment, additional buffer space is required based on the following calculation:

CAPTBUFF value x 2

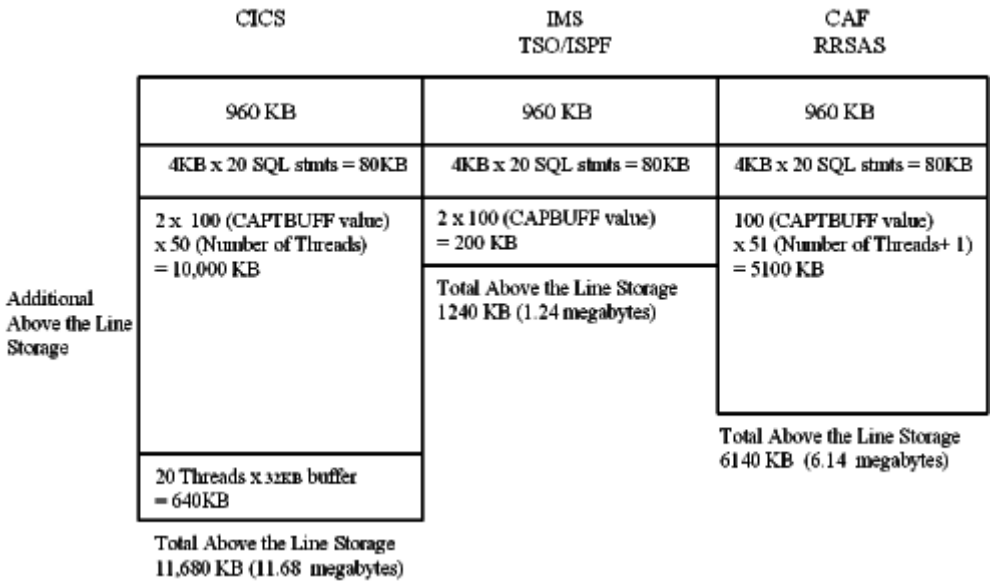
Call Attachment Facility (CAF) and RRSAS Environments

Beside the storage space required for every environment, additional buffer space is required based on the following calculation:

(CAPTBUFF value x (Number of Threads +1))

Figure 2-3 on page 2-4 shows how much above the line storage is required by the *DB2 Feature* if the DB2 system has 50 threads running, the CAPTBUFF parmeter value is set at 100, and STROBE samples 20 unique SQL statements:

Figure 2-3. DB2 Multi-Thread Above the Line Storage Requirements



Suppressing Statement Count and Service Time Information

If you do *not* want to collect statement count and service time information, complete the following steps:

1. Select Option 1 from the STROBE OPTIONS menu. STROBE/ISPF displays the STROBE - ADD ACTIVE REQUEST panel (Figure 2-1 on page 2-2)
2. Enter "Y" in the DATA COLLECTORS field and press **Enter**. STROBE /ISPF displays the DATA COLLECTORS panel shown in Figure 2-4 on page 2-5.
3. Enter "N" in the DB2 field of the CAPTURE Options section on the STROBE-DATA COLLECTORS panel.
4. Press **Enter** to submit the measurement request.

Note: Complete these steps only if you do not wish to collect SQL statement count and service time.

Figure 2-4. STROBE - DATA COLLECTORS Panel

```
--------------------------- STROBE - DATA COLLECTORS --------------------------
 COMMAND ===>

 OVERRIDE DATA COLLECTOR DEFAULTS FOR JOBNAME:  WPAJEA
 DATA COLLECTORS: (Y or N; Y adds to and N removes from your system defaults)
   ADABAS           ===>     ADA3GL           ===>     C                ===>
   CICS             ===>     COBOL            ===>     CSP              ===>
   DB2              ===>  Y  IDMS             ===>     IDMS BATCH DML   ===>
   IEF              ===>     IMS              ===>     JAVA             ===>
   MQSERIES         ===>     NATURAL          ===>     PL/I             ===>
   SVC              ===>

 CICS Options:
   Collect Region Data ===>  Y       OR   Produce Performance Supplement ===>
   Detail Transaction (TRAN or TR*):          Collect Terminal Activity ===>
   =>      =>       =>      =>      =>

 CAPTURE Options: (Y or N; default is Y)
            DB2    ===>    IMS    ===>

 MQ Common User Module    ===>              Always use as default (Y/N)  ===>

 OTHER DATA COLLECTORS:
   PROGRAM NAME   ===>           ===>           ===>           ===>
```

With STROBE command language, specify the DB2=NOCAPTURE parameter. The capture facility will be disabled for this session of STROBE only. The next time STROBE starts, information on statement count and service time will be gathered unless you disable it as just described.

Disabling the Collection of DB2 Data

Every time a measurement request for a DB2 application is issued, the STROBE DB2 Feature starts. If you want to disable the STROBE DB2 Feature, you must specify that it should not start.

To set these parameters with STROBE/ISPF:

1. Enter “Y” in the DATA COLLECTORS field of the STROBE - ADD ACTIVE or STROBE - ADD QUEUED panel.
2. Enter “N” in the DB2 field on the STROBE - DATA COLLECTORS panel (Figure 2-4).

With STROBE command language, specify the NODB2 operand when you submit an ADD command.

Note: Specifying NODB2 disables all DB2 data collection, including SQL attribution, QMF tracing, SQL statement count and service time information.

Disabling the STROBE DB2 Feature QMF Trace Attributor

The STROBE DB2 Feature uses the QMF Trace Attributor to determine the names of any QMF Procedures and Queries that are active during the measurement session, as described in the section titled “Reporting QMF Procedure and Query Structures” on page 3-21. If you disable the Trace function, QMF Procedures, and some Queries with low activity levels will not be reported in the Performance Profile. You can disable the QMF Trace Attributor by specifying corresponding parameters, as described below.

To set these parameters with STROBE/ISPF:

1. Enter “Y” in the OTHER PARAMETERS field of the STROBE - ADD ACTIVE or STROBE - ADD QUEUED panel.
2. Specify, on the STROBE - OTHER PARAMETERS panel:

3. DB2=NOQMFTRACE

With STROBE command language, specify the DB2=NOQMFTRACE operand when you submit an ADD command.

Creating the STROBE Performance Profile

This section explains several parameters you may want to set depending on the type of information you need to analyze your DB2 environment. STROBE enables you to customize how the Performance Profile reports DB2 measurement data so you can aggregate and suppress information about your DB2 environment according to your needs and interests. Chapters 3 and 6 in the *STROBE MVS User's Guide* and Chapters 3 and 7 in the *STROBE MVS User's Guide with Advanced Session Management* explain other details about how to produce the Performance Profile.

If you are using the STROBE CICS Feature to measure the performance of CICS transactions that are accessing DB2, you will want to use specific CICS reports that can give you a more complete understanding of your DB2 environment. Refer to the STROBE *CICS Feature* for information about using the CICS Performance Supplement and the transaction reports it provides.

Activity Aggregation and Expansion

Depending on the type of activity it detects, STROBE can either aggregate or provide more detail by expanding the breadth of the information it displays in the Performance Profile. When you examine the reports detailing DB2 activity, you need to know exactly what is represented by the values shown in the report fields, or you might incorrectly view how your code is running. Use the following guidelines when examining the reports:

DBRMs

The STROBE DB2 Feature *always* aggregates activity in DBRMs that have the same name and creation timestamp or package version number.

SPUFI Files

The STROBE DB2 Feature does not aggregate activity in SPUFI files.

Procedures and Queries

The STROBE DB2 Feature automatically combines the CPU percentage from multiple executions of a single Query or Procedure into a report that shows the aggregate CPU time for SQL statements across every execution of the QMF Query or Procedure. STROBE aggregates Queries when they have the same name and identical SQL text. It aggregates Procedures when they invoke the same Queries, Procedures, and QMF commands in the same order, and when all invoked Queries and Procedures are unchanged.

When calculating totals, STROBE groups

- all stand-alone executions of a Query
- all stand-alone executions of a Procedure, including one with Subprocedures
- *consecutive* invocations of a single Query or first-level Subprocedure within the same Procedure (STROBE does *not* aggregate any Subprocedures below first-level)

When procedures invoke Queries, other Procedures and QMF commands, the STROBE DB2 Feature displays the structure of the procedure and shows the order in which they were invoked. When other procedures are invoked as subprocedures, STROBE shows whatever statement or procedure they invoke. See the section called "Reporting QMF

Procedure and Query Structures" on page 3-21 for a complete explanation of how these activities are reported.

Procedure and Query Identification and Reporting

STROBE provides several approaches you can use to control how STROBE reports on Procedures, Queries, and SPUFI files. You can

- cause STROBE to report each execution of a Query or Procedure as a unique entity rather than as an aggregate of a calling Procedure
- suppress SQL Procedure reports without suppressing the Query reports

You could also place a group of Queries for which you want an aggregated total in a single Procedure and STROBE would report one total value for all of the queries.

Separating Procedures and Queries into Individual Entities

The STROBE DB2 Feature automatically combines activity for all executions of a particular Procedure or Query into one entity. You can see detail for each execution of a Procedure or Query.

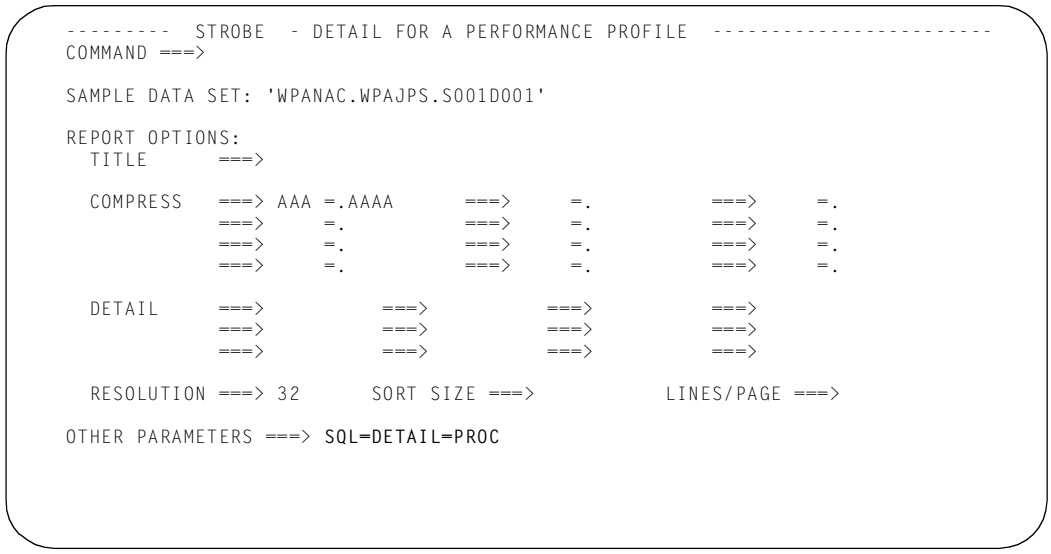
To set these parameters with STROBE/ISPF:

1. Enter "Y" in the Detail Reports field of the STROBE - PRODUCE A PERFORMANCE PROFILE panel.
2. In the OTHER PARAMETERS field on the STROBE -DETAIL FOR A PERFORMANCE PROFILE panel (Figure 2-5 on page 2-8), specify:
   - for detailed reports of Procedures only, `SQL=DETAIL=PROC`
   - for detailed reports of Queries only, `SQL=DETAIL=QUERY`
   - for detail reports of both Procedures and Queries, `SQL=DETAIL=(PROC,QUERY)`

With the STROE or STROXE procedure, specify the SQL=DETAIL={PROC|QUERY|(PROC,QUERY)} option.

For a complete list of the options that you can specify in the OTHER PARAMETERS field, see Chapter 3 of the *STROBE MVS User's Guide* or *STROBE MVS User's Guide with Advanced Session Management*.

Figure 2-5. STROBE - DETAIL FOR A PERFORMANCE PROFILE Panel



SQL Text Reporting

The STROBE DB2 Feature automatically records the text of the SQL statements that are active during the measurement session. If STROBE cannot identify a statement's text, the Performance Profile shows the statement's number and type. The STROBE DB2 Feature does not provide SQL text for inactive or nonexecutable SQL statements.

Controlling SQL Statement Text Reporting

You can control when STROBE should display the entire text of a SQL statement in the Performance Profile by specifying thresholds with two parameters called CPUTEXT and WAITTEXT. The parameter values determine whether a SQL statement has caused enough CPU usage or wait time to appear in either the CPU Usage by SQL Statement or the Wait by SQL Statement reports.

For example, if the CPUTEXT parameter were set to 25.0, full SQL text will be printed for all SQL statements that consume more than 25 percent of the CPU time for a measurement session. If the full text of a SQL statement does not appear in the Performance Profile report in which you are interested, you should decrease the value for either CPUTEXT or WAITTEXT and re-create the Profile. The default values are 10% for CPUTEXT and 5% for WAITTEXT.

Note: For static SQL, a limitation imposed by DB2 prevents STROBE from reporting more than 4 kilobytes of text.

To set these parameters with STROBE/ISPF:

1. Enter "Y" in the Detail Reports field of the STROBE - PRODUCE A PERFORMANCE PROFILE panel.
2. In the OTHER PARAMETERS field on the STROBE - DETAIL FOR A PERFORMANCE PROFILE panel, specify: SQL=CPUTEXT= *nn.n* where *nn.n* equals the percentage of CPU time consumed, or SQL=WAITTEXT=*nn.n*, where nn.n equals the percentage of wait time observed for a SQL statement during a measurement session. (You can specify both CPUTEXT and WAITTEXT.) Any value for a SQL statement greater than the percentage specified will cause STROBE to report all of the available SQL text for the statement.

For statements that do not exceed the thresholds specified by either CPUTEXT or WAITTEXT, STROBE truncates the text to a specified number of characters.

- The MAXLEN=*nnnn* operand controls the number of characters STROBE prints for SQL statements that fall below the threshold values you specified. You must specify a value between 0 and 9999 (default **300**) for *nnnn*.

  Note: If you enter a value that is *not* a multiple of 100, STROBE rounds up to the next hundred.

- To suppress the SQL text for statements that do not exceed the specified thresholds , enter "0".
- To print all available text for all SQL statements, specify "0" for CPUTEXT, "0" for WAITTEXT and "9999" for MAXLEN.

With the STROE or STROXE procedure, specify the SQL=MAXLEN=*nnnn* option or SQL=CPUTEXT=*nn.n* or SQL=WAITTEXT=*nn.n* options.

Attribution Reporting

To limit or suppress the Attribution of CPU Execution Time and Wait Time reports in the Performance Profile, specify corresponding parameters on the STROBE - TAILOR REPORTS panel, as shown in Figure 2-6 on page 2-10.

Compressing Attribution Reports with Low Activity

You can compress attribution reports for any system service module in which the total CPU or wait time percentage is less than a specified baseline. By default, STROBE will not report on any service module that has consumed less than 2% of the CPU usage during a measurement session, or waits less than 2% of the time of the session. You can change this value through either using ISPF or the STROBE command language.

To set these parameters with STROBE/ISPF:

1. Enter "Y" in the Tailor Reports field of the STROBE - PRODUCE A PERFORMANCE PROFILE panel.
2. On the STROBE - TAILOR REPORTS panel, specify a baseline percentage between 0 and 99.9 in the "Compress below %" portion of the ATTRIBUTION REPORTS field (Figure 2-6 on page 2-10).

When you submit a batch job using the STROE or STROXE procedure, specify ATTR=*nn.n*.

Suppressing All Attribution Reports

You can suppress *all* Attribution reports.

To set these parameters with STROBE/ISPF:

1. Enter "Y" in the Tailor Reports field of the STROBE - PRODUCE A PERFORMANCE PROFILE panel.
2. On the STROBE - TAILOR REPORTS panel, enter "Y" in the "Suppress reports for" portion of the ATTRIBUTION REPORTS field (Figure 2-6 on page 2-10).

With the STROE or STROXE procedure, specify the NOATTR parameter.

Figure 2-6. STROBE - TAILOR REPORTS Panel

```
--------------------------- STROBE - TAILOR REPORTS ---------------------------
COMMAND ===>

WAIT TIME BY MODULE -- Show location of wait  ===>   (Specify Y)
-------------- Report --------------- Compress below   OR  Suppress (Specify Y)
PROGRAM USAGE BY PROCEDURE                ===>        %         ===>
DASD USAGE BY CYLINDER                    ===> 02.0   %         ===>
TRANSACTION USAGE BY CONTROL SECTION      ===>        %         ===>
CICS TRANSACTION PROFILE                  ===>        sec       ===>
CICS REGION LEVEL                         ===>        sec       ===>
MQSERIES CALLS                            ===>        %
ATTRIBUTION Reports                       ===>        %         ===>  Y
  Suppress reports for:
        C         ===>       CICS      ===>       COBOL     ===>
        CSP       ===>       DB2       ===> Y     DL/I      ===>
        IDMS      ===>       IEF       ===>       MQSERIES  ===>
        PL/I      ===>       SVC       ===>                 ===>
PROGRAM SECTION USAGE SUMMARY                      Display inactive  ===>
TIME and RESOURCE DEMAND DISTRIBUTION
 Combine tasks ===>    Display all tasks ===>    Display all DDs   ===>

CICS TRANSACTION PROFILE       FILTERS =>      =>      =>      =>      =>
 Suppress non-CICS TRANSACTION REPORTS ===>   (Specify Y)
USE DATE AND TIME FORMAT FROM PARMLIB  ===>
```

Suppressing DB2-Specific Attribution Reports

You can suppress the Attribution of CPU Execution and Wait Time reports for DB2 service modules *only*.

To set these parameters with STROBE/ISPF:

1. Enter “Y” in the Tailor Reports of the STROBE - PRODUCE A PERFORMANCE PROFILE panel.
2. On the STROBE - TAILOR REPORTS panel, enter “Y” in the Suppress reports for DB2 field (Figure 2-6).

With the STROE or STROXE procedure, specify the NOATTR=DB2 operand.

Suppressing Procedure Reporting

You can suppress Procedure reports in the Performance Profile.

To set these parameters with STROBE/ISPF:

1. Enter “Y” in the Detail Reports field of the STROBE - PRODUCE A PERFORMANCE PROFILE panel.
2. In the OTHER PARAMETERS field on the STROBE - DETAIL FOR A PERFORMANCE PROFILE panel, specify:

   ```
   SQL=NOPROC
   ```

With the STROE or STROXE procedure, specify the SQL=NOPROC operand.

Note: The SQL=DETAIL=PROC option (discussed in the section titled “Separating Procedures and Queries into Individual Entities” on page 2-7) and the NOPROC option are mutually exclusive.

Chapter 3.
The Performance Profile for a DB2 Allied Address Space

This chapter describes the STROBE Performance Profile produced when you measure a DB2 allied address space. It includes an overview of DB2 batch reports and describes specific reports associated with QMF and SPUFI address spaces.

Choosing the reports you need to examine is dependent on the characteristics of your DB2 environment and where you want to improve performance. For example, you might want to focus on DB2 system modules or on a specific DBRM; or an overall reduction in CPU usage may be your goal.

Some reports show CPU usage. Another set of reports shows the time a job waits. You can set various reporting parameters to customize what data appears in your Performance Profile. This information can provide a comprehensive picture of all factors influencing how your application is running. For a complete description of the STROBE Performance Profile, see Chapter 3 of STROBE *MVS Concepts and Facilities.*

Reports That Show CPU Execution

The Performance Profile shows CPU execution in DB2 allied address spaces in four types of reports:

- Program module reports
- Attribution reports
- SQL, DBRM, Query, and Procedure reports
- Transaction reports for batch jobs

Each type of report shows a different view of CPU execution activity during a measurement session. You can use various combinations of the reports to trace the cause of heavy CPU usage. After each type of report is briefly described, examples of these report combinations are provided to show how to follow the report findings to locate the source of the CPU consumption.

Program Module Reports

The Program Section Usage Summary and Program Usage by Procedure reports show CPU usage in all programs, including application programs, DB2 system modules, and other system service routines. The Program Section Usage Summary report shows where the majority of the CPU usage occurred: in DB2 service modules or user programs. After you identify where the heavy CPU usage is taking place, refer to the Program Usage by Procedure report for these program sections to see more specifically what is responsible for the CPU usage.

The Most Intensively Executed Procedures report highlights the information in the module reports. It lists the ten heaviest users of CPU time and quickly shows you whether service modules or user programs are responsible for a majority of the CPU usage.

Throughout the STROBE Performance Profile, STROBE reports activity in DB2 system modules to pseudo-sections within the pseudo-module .SYSTEM, as shown below.

Module Name Prefix	Pseudo-Section	Function
DSN	.DB2	DB2 system services
DSQ	.QMF	Query management
DXR	.IRLM	DB2 resource locking

You can obtain detailed reporting on activity within a DB2 module or control section by specifying the module name in the DETAIL parameter when you generate the Performance Profile. For more information, see Chapters 2 and 4 of the STROBE *MVS User's Guide*. For more information on STROBE pseudo-entities, see Appendix A of STROBE *MVS Concepts and Facilities*.

Attribution Reports

Attribution of CPU Execution Time reports for batch programs identify the callers of DB2 system modules by transaction name and by

- SQL statement within DBRM, or
- SQL statement within Query

When a DB2 service module shows high CPU usage, its Attribution of CPU Execution Time report identifies all statements that invoked the service.

SQL, DBRM, Query, and Procedure Reports

The SQL CPU Usage Summary report shows which DBRMs, Queries, and Procedures were responsible for CPU activity. The CPU Usage by SQL Statement reports show the distribution of the CPU activity among the SQL statements that make up those DBRMs, Queries, and Procedures. Turn to these reports when most of the CPU activity occurs in DB2 service modules.

Transaction Reports

The Transaction Summary and Transaction Usage by Control Section reports identify which transactions are responsible for CPU activity in online address spaces. These reports are helpful when

- you want to see transaction and CPU usage in CICS response time statistics
- DB2 service modules do *not* account for a majority of the CPU activity
- you want to see plan names for batch jobs

Reports That Show Wait Time

Four STROBE reports show wait time:

- The Wait Time by Module report shows all the modules, control sections, pseudo-modules and pseudo-sections in which STROBE found the application it was measuring in wait state.
- The Attribution of CPU Wait Time identifies the percentage of wait time spent in an invoked system routine as a result of the routine that invoked it.

- The SQL Wait Time Summary report shows the distribution of wait time among Queries, Procedures, and DBRMs.
- The Wait Time by SQL Statement report shows the distribution of wait time among SQL statements in a Procedure, Query, or DBRM.

Choosing Between the Execution and Wait Reports

To determine whether to examine the CPU execution or wait reports in the Performance Profile, determine from the Measurement Session Data report (Figure 3-1) whether the value shown for CPS TIME PERCENT (the percentage of time that one or more CPUs were active) is greater than the value shown for WAIT TIME PERCENT, in the MEASUREMENT STATISTICS section of the report.

Figure 3-1. Measurement Session Data Report for a Batch Job

```
                                    ** MEASUREMENT SESSION DATA **

------- JOB ENVIRONMENT --------        ------- MEASUREMENT PARAMETERS ------        ----- MEASUREMENT STATISTICS ----

PROGRAM MEASURED   -     IKJEFT01         ESTIMATED SESSION TIME  -      10 MIN       CPS TIME PERCENT  -          94.46
JOB NAME           -     WPALXMA3         TARGET SAMPLE SIZE      -      10,000       WAIT TIME PERCENT -           5.54
JOB NUMBER         -     JOB32090         REQUEST NUMBER (Q)      -          79       RUN MARGIN OF ERROR PERCENT -  .98
STEP NAME   -            PH02CS04                                                     CPU MARGIN OF ERROR PERCENT -  .98

                                          SYS REQ  -  SCS01                         TOTAL SAMPLES TAKEN -        1,497
DATE OF SESSION    -     09/08/99                                                   TOTAL SAMPLES PROCESSED -    1,497
TIME OF SESSION    -     15:19:45                                                   INITIAL SAMPLING RATE  - 16.67/SEC
                                        --------- REPORT PARAMETERS ---------      FINAL SAMPLING RATE   -  16.67/SEC

SYSTEM -               ESA SP4.3.0     REPORT RESOLUTION   -        32 BYTES      SESSION TIME -     4 MIN 50.98 SEC
DFSMS     -                 1.1.0      SORTSIZE      -               999,999       CPU TIME      -    0 MIN 32.04 SEC
SUBSYSTEM  -             DB2 3.1.0      LINES/PAGE    -                   60       WAIT TIME     -    0 MIN  1.60 SEC
DB2SUBSYSTEMID-               WPA1      DASD= 2.0%     DASDGAP=5    ATTR=0.0%
DB2 APPLICATION            DB2COPYD
CPU MODEL    -            3090-600S                                               STRETCH TIME -     4 MIN  17.34 SEC
SMF/SYSTEM ID-
           SESA/SCS01

                                                                                  SRB TIME     -     0 MIN   0.43 SEC
REGION SIZE BELOW 16M  -    1,088K                                                SERVICE UNITS  -             30951
REGION SIZE ABOVE  -       32,768K                                                PAGES IN -     0     OUT-         0
                                                                                  PAGING RATE  -            0.00/SEC
PTF LEVEL 2.3.0. FS000000/FS000000                                                EXCPS -         95        0.33/SEC
STROBE TAPE NUMBER      000/S00DSK

SAMPLE DATA SET - ZZ.WPALXMA3.S002D001
```

The report shows the CPS time percent is 94.46. To reduce CPU time consumption, first determine what is using CPU time by examining the Program Section Usage Summary and the Transaction reports. If you determine that the majority of time is being consumed by application programs, the STROBE *MVS Concepts and Facilities* provides a detailed example of how to use the Performance Profile to reduce application program CPU usage.

If STROBE finds a significant amount of wait time, see"Reports for Analyzing Wait Time" on page 3-14.

CPU Reports for Analyzing Batch Jobs

This section describes Performance Profile reports you can use to determine which parts of your application are consuming the most CPU time and if improvement can be made in the application itself, its utilization of system resources, or in both areas.

Program Section Usage Summary Report

The Program Section Usage Summary report (Figure 3-2) shows the distribution of CPU time used by each active control section of each active module in the DB2 address space. From it, you can determine whether DB2 system modules or user programs are responsible for most of the CPU activity.

The report shows that DB2 system modules used 87.27% of the execution time. STROBE compresses DB2 and other IBM system module information and shows pseudo-sections under the pseudo-module .SYSTEM as highlighted on the example below.

When DB2 system modules are the heaviest CPU users, you should refer to either the SQL CPU usage reports to see which SQL statements are affecting performance or the Program Usage by Procedure report to see what system functions DB2 is using. If user programs show high activity, also refer to the Program Usage by Procedure reports.

Figure 3-2. Program Section Usage Summary Report

```
                                    ** PROGRAM SECTION USAGE SUMMARY **
MODULE  SECTION 16M   SECT FUNCTION                  CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  2.61%
   NAME  NAME    <,>   SIZE                            SOLO     TOTAL        .00     22.00     44.00     66.00     88.00

.SYSTEM  .DB2               DB2 SYSTEM SERVICES       87.27     87.27        .**************************************
.SYSTEM  .IRLM              RESOURCE LOCK MANAGER       .99       .99        .
.SYSTEM  .NUCLEUS           MVS NUCLEUS                6.01      6.01        .**
.SYSTEM  .SVC               SUPERVISOR CONTROL          .14       .21        .
.SYSTEM  .XMEMORY           CROSS MEMORY               1.41      1.41        .
                                                      -----     -----
.SYSTEM  TOTALS             SYSTEM SERVICES           95.82     95.89

IGC0004B          >    7688 SUPERVISOR SERVICES         .07       .07        .
IRARMFIP          >    4640 SYSTEM RESOURCE MGMT       1.63      1.63        .
IRARMGLU          <     240 SYSTEM RESOURCE MGMT        .14       .14        .

TESTBC3  .DB2          5856 DB2 SYSTEM SERVICES        1.77      1.77        .
TESTBC3  TESTBC3      31072                             .50       .50        .
                                                      -----     -----
TESTBC3  TOTALS   <   36928                            2.27      2.27
                                                      -----     -----
PROGRAM IKJEFT01 TOTALS                               99.93    100.00
```

Program Usage By Procedure Reports

The Program Usage by Procedure reports present a detailed accounting of CPU time spent by each Procedure within each program that was active during a measurement session. The format of an application program report differs slightly from the report format for system modules.

- For application programs, STROBE displays the module name, control section name, and starting location. If the control section was indexed, STROBE shows the line number and Procedure name as well.
- For system modules, STROBE displays the module and control section name and a function descriptor. Both types of reports show the solo and total CPU time used by each control section within the module. (For a complete description of this report, see Chapter 3 in STROBE *MVS Concepts and Facilities*.)

Activity in System Modules

If the Program Usage Summary report indicates that DB2 system modules accounted for most of the CPU activity, the Program Usage by Procedure in Figure 3-3 on page 3-5 shows the DB2 modules STROBE found to be utilized during the measurement session.

Figure 3-3. Program Usage by Procedure Report for DB2 Modules

```
                                                ** PROGRAM USAGE BY PROCEDURE **
          .SYSTEM     SYSTEM SERVICES                          .DB2      DB2 SYSTEM SERVICES
MODULE   SECTION        FUNCTION             INTERVAL  CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  1.28%
   NAME      NAME                              LENGTH    SOLO     TOTAL        .00      6.00     12.00     18.00     24.00

 DSNAPRH             PGM REQUEST APPL INTERF       504     .19       .20        .
 DSNB1CLM            CLAIM PGSET/PART             6880     .07       .09        .
 DSNB1CMU            BUFFER MGR COMMIT CSECT      3848     .02       .02        .
 DSNB1CPS            CLOSE PG SET                 7760     .03       .03        .
 DSNB1GET            RETRIEVE REQUESTED PAGE     12832    1.74      1.82        .**+
 DSNB1OFA            LOCATE PAGSET BLOCK (PB      3768     .03       .05        .
 DSNB1REL            PAGE RELEASE ROUTINE         7432     .66      1.18        .*
 DSNB1SPF            SCH NORML PFTCH IN SEQ       5840     .05       .05        .
 DSNCEXT1            CICS ATTACHMNT EXITS         7176    1.31      1.33        .**
 DSNCEXT3            CONNECTION SUBTASK           4576     .36       .39        .
 DSNCSM31            CICS ATTACH MODULE 3.1       2584     .55       .55        .
 DSNGEDM             DATA MGT, DBD/SKCT RTNS    173096     .12       .12        .
 DSNIDM   DSNIBCTD   LOCATE/CREATE A CTDB         5392     .17       .17        .
 DSNIDM   DSNICMT2   DM COMMIT PHASE2 PROCED     11200     .02       .02        .
 ...
 DSNIDM   DSNIMSAR   RELEASE ALL MSA RESOURC       2664     .05       .05        .
 DSNIDM   DSNIMSCU   RECORD OF MSA TO A CUB       4752     .03       .03        .
 DSNIDM   DSNIMSMS   GET NEW REC,RLSE OLD RE      5240     .03       .03        .
 DSNIDM   DSNINXTP   READ SPECFD PAG USNG MS      4968     .41       .46        .
 DSNIDM   DSNIOSET   WRT TRACE FOR IX SCAN        32112    3.69      3.78        .*****+
 DSNIDM   DSNIPSFI   LOCATE/CREATE FILE PSCB      11000     .12       .14        .
 DSNIDM   DSNIPSI    LOCATE/CREATE INDEX PSC      6696     .03       .05        .
 DSNIDM   DSNIREDR   EXECUTE DB PROCEDURES        3216     .98      1.07        .*
 DSNIDM   DSNIRIDL   LOOP, ENTR RID IN RIDLS      7608     .07       .09        .
 DSNIDM   DSNIRIDR   IX ONLY, RETRV RID LIST      8488     .05       .05        .
 DSNIDM   DSNIRNXT   FETCH NEXT ROW TO PROG      25952     .46       .49        .
 DSNIDM   DSNIRSET   SET A CUB BY RID            12344     .03       .03        .
 DSNIDM   DSNISELK   EVAL SELECTION EXPRESIO      4104     .02       .02        .
 DSNIDM   DSNISFED   EVAL SET FUNCTS WITH RE      6640   43.45     43.45        .++++++++++++++++++++++++++++++++++++
 DSNIDM   DSNISFS    TS, EVAL SET FUNS FR GR     14024    6.71      6.75        .++++++++++
 DSNIDM   DSNISM     DATA MGR'S STOR MGR          1896     .09       .09        .
 DSNIDM   DSNISRID   SET CUB BY LST OF RIDS      18168     .20       .22        .
 DSNIDM   DSNITCUS   MSI  SET CUB STAT=BEFOR      7824     .15       .17        .
 DSNIDM   DSNIZLDE   DECOMPRES DB2 ROW, Z-L        848     .17       .19        .
 DSNKDM              DM INDEX MNGMNT FUNCTIO    339072     .46       .49        .
 DSNRGLM1            RMC IEPL MODULE/GLOBAL      24656     .03       .03        .
 ...
 DSN3EPX  DSNAPRHX                                5432    1.69      1.72        .**
 DSN3EPX  DSN3AA                                   256     .44       .55        .
 DSN8EAE1                                          504    1.70      1.74        .**
                                                          -----     -----
          .DB2       TOTALS                              87.26     87.27
```

As the highlighted area of the report shows, DB2 section DSNISFED of module DSNIDM, accounts for 43.45% of the CPU activity. To determine which transactions and SQL statements caused the activity, examine the Attribution reports for this module and control section.

Attribution of CPU Execution Time Report

The Attribution of CPU Execution Time report (Figure 3-4 on page 3-6) identifies the sites of invocation of system routines. Use this report to identify which SQL statements caused activity in a system service routine. This report contains the following sections:

- Header lines
- Detail lines
- Total line

Figure 3-4. Attribution of CPU Execution Time Report

```
                                    ** ATTRIBUTION OF CPU EXECUTION TIME **
.DB2     DSNISFED           AGNT SERV/LATCH MGT DRVR
 -------------------------WAS INVOKED BY--------EXECUTING-------------------TARGET-----------------------  CPU TIME %
 XACTION  QUERY NAME                  TIME       STMT TEXT                   STMT TEXT                     SOLO  TOTAL

 TESTBH31 TESTBC3                     12:58:05   1159 FETCH                    01 DECLARE                   .07  .07
                                                                                                          ----- -----
                                                                                                           .07  .07
.DB2     DSNVRMEL           RESOURC MGR LINKAGE CNTL
 -------------------------WAS INVOKED BY--------EXECUTING-------------------TARGET-----------------------  CPU TIME %
 XACTION  QUERY NAME                  TIME       STMT TEXT                   STMT TEXT                     SOLO  TOTAL

 TESTBH31 TESTBC3                     12:58:05   1954 FETCH                    03 DECLARE                  1.13  1.13
                                                                                                          ----- -----
                                                                                                          1.13  1.13
.DB2     DSNXEEZ  DSNXEEZ   DATA BASE PROGRM REQUEST
 -------------------------WAS INVOKED BY--------EXECUTING-------------------TARGET-----------------------  CPU TIME %
 XACTION  QUERY NAME                  TIME       STMT TEXT                   STMT TEXT                     SOLO  TOTAL

 TESTBH31 TESTBC3                     12:58:05   1954 FETCH                    03 DECLARE                  3.82  3.82
                                                                                                          ----- -----
                                                                                                          3.82  3.82
.DB2     DSNXGRDS DSNXEBR   LINK CSECT RDS/ACCES MOD
 -------------------------WAS INVOKED BY--------EXECUTING-------------------TARGET-----------------------  CPU TIME %
 XACTION  QUERY NAME                  TIME       STMT TEXT                   STMT TEXT                     SOLO  TOTAL

 TESTBH31 TESTBC3                     12:58:05   1954 FETCH                    03 DECLARE                  2.90  2.90
                                                                                                          ----- -----
                                                                                                          2.90  2.90
.DB2     DSNXGRDS DSNXECP   COPY APPLCTN STRUCTURES
 -------------------------WAS INVOKED BY--------EXECUTING-------------------TARGET-----------------------  CPU TIME %
 XACTION  QUERY NAME                  TIME       STMT TEXT                   STMT TEXT                     SOLO  TOTAL

 TESTBH31 TESTBC3                     12:58:05   1954 FETCH                    03 DECLARE                  6.58  6.58
                                                                                                          ----- -----
                                                                                                          6.58  6.58
```

Header Lines

The report header identifies the invoked routine, showing

- its pseudo-module, module, and control section name (when available)
- a function descriptor for either the control section or the module

Detail Lines

Each report detail line for a DB2 service routine identifies the SQL statement that invoked the module and displays

- a transaction name, if one is available
- the name of the SQL statement, DBRM, or Query
- the time that the module was created
- its SQL statement number (if available)
- the executed SQL statement's type and the target statement's number and type
- the solo and total CPU time spent on behalf of the invoker

Total Line

The total line shows the total time attributed to the invokers of the module. It may be less than the time shown in the Program Usage by Procedure or Most Intensively Executed Procedures reports because STROBE cannot always identify an invoker of a service routine.

If you want to examine an SQL statement on the Attribution of CPU Execution Time Report in more detail, see "CPU Usage by SQL Statement Report" on page 3-9.

CPU Usage Reports for SQL

Instead of beginning with the Program Usage by Procedure reports, you can start your analysis with the SQL CPU Usage Summary report. It presents an overview of the CPU activity for each DBRM, Query, or Procedure. For DBRMs, the report also shows total statement counts. The CPU Usage by SQL Statement reports show a detailed account of the CPU usage by SQL statement. This report displays statement counts and the CPU percents for each SQL statement in a Query, Procedure, or DBRM that was active during measurement. See "Reports Customized for QMF and SPUFI" on page 3-17 for report examples showing Queries and Procedures.

Note: QMF owns several DBRMs (whose names begin with "DSQ"), which it uses for initialization and SQL processing. Because these DBRMs are associated with QMF rather than the invoking SQL statement, STROBE always attributes the CPU activity in these DBRMs to QMF. These DBRMs are listed as separate elements in the SQL usage reports.

SQL CPU Usage Summary Report

The SQL CPU Usage Summary report shown in Figure 3-5 on page 3-8 shows the distribution of CPU activity among DBRMs, queries, and procedures that invoked DB2 system services. Each detail line in this report shows the executing SQL type, its name, the time and date it was executed (or, for DBRMs, the time and date it was created), and its solo and total CPU usage. Also for DBRMs, STROBE provides the total executed statements for each DBRM and the average amount of execution time for all of the statements.

The following limitations apply to STROBE SQL statement count and Execution time reporting:

- STROBE does not begin to collect counts and service time data for an SQL statement until it takes a sample in the statement. An SQL statement could execute several times before STROBE takes a measurement sample. Once the first sample is taken during a SQL statement execution, STROBE will increment the statement count and adjust the service time for each subsequent run of the statement whether or not it is sampled. A higher sampling rate will produce more complete counts.
- For active requests, STROBE cannot report count or execution information on an instance of a SQL statement processing at the time of the request. STROBE will report on subsequent executions of the statement.
- For queued requests, STROBE may not collect a count and service time for the first execution of a SQL statement if the program does not make an explicit connection to DB2 before it issues the statement. STROBE will begin collecting a count and service time for all subsequent statements.
- To increment the count and adjust the service time of dynamic non-cursor SQL statements, STROBE must have measured the initial PREPARE statement to correlate it with the corresponding subsequent EXECUTE statement. A higher sampling rate will produce more complete counts.
- To increment the count and adjust the service time of dynamic cursor SQL statements, STROBE must have measured the PREPARE and OPEN statements of the cursor to correlate FETCH statement(s) to the SELECT statements. A higher sampling rate will produce more complete counts.
- When execution time is negligible for a SQL statement, STROBE rounds the amount to zero and a service time of .0000 is reported.

Figure 3-5. SQL CPU Usage Summary Report

```
                                          ** SQL CPU USAGE SUMMARY **
SQL    SQL             STMT-EXECUTION    TIME/COUNT          % CPU TIME        CPU TIME HISTOGRAM  MARGIN OF ERROR:  1.28%
TYPE   NAME             CNT  AVG TIME                      SOLO    TOTAL       .00     12.50     25.00     37.50     50.00

DBRM  SQLDYNAM         5734     .0244   16:13:36 23JUN99  46.22   46.53       .*********************************
DBRM  SQLDYNM1          492     .1392   16:13:06 23JUN99  11.43   13.53       .********++
DBRM  SQLDYNM2         5270     .0206   16:12:38 23JUN99  16.24   22.69       .************++++++
                                                          -----   -----
      TOTAL SQL CPU USAGE                                73.89   82.75
```

SQL TYPE

In the SQL TYPE column of the report, the STROBE DB2 Feature abbreviates the type of the executing SQL application. The following abbreviations denote the types of SQL encountered during the measurement session.

STROBE Abbreviation	Complete SQL Type
DBRM	Database request module
CMD	QMF command
QSAV	Saved Query
PRM	Prompted Query
QBE	Query by example
SQL	Interactive SQL
QRY	Query invoked by a Procedure
SPUF	SPUFI input file
QPROC	QMF Procedure
SUBPROC	An embedded QMF Procedure

STROBE displays the applications in the following order, regardless of the actual order of execution: DBRM, SPUFI files, Saved Queries, Prompted Queries, Queries by Example, Interactive SQL statements, QMF commands, and Procedures. Within each category, the applications are sorted first by name and second by time and date. (If the Query does not have a name, the sort order is unpredictable.) For example, if two Saved Queries named CHKNAME contain different SQL statements and are executed at 08:25 and 10:15, respectively, STROBE displays two separate detail lines for CHKNAME and reports first the information for the execution at 08:25.

SQL NAME

In this column, STROBE displays the actual DBRM, package, procedure or query name.

STMT CNT

In this column, STROBE displays the total number of statement executions during the measurement session. This information only appears for DBRMs and SQL executed by the Remote Resource Services Attachment Facility (RRSAF).

EXECUTION AVG TIME

In this column, STROBE displays a weighted average for the number of seconds it took for all statements to execute. This information only appears for DBRMs and SQL executed by the Remote Resource Services Attachment Facility (RRSAF).

Note: If you are using the Call Attachment Facility and a DSNTRACE DD statement is present in the job you are measuring, the statement count and execution average times reported by STROBE will be inaccurate.

TIME/COUNT

STROBE determines the contents of this column from the DBRM, Query or Procedure type, the number of times they executed, and the SQL options the user specified when generating the Performance Profile. Depending on what STROBE is measuring, the column displays different kinds of information.

- For a DBRM, STROBE displays the time and date the DBRM was created.
- If the Query was executed once or SQL=DETAIL=QUERY was specified, STROBE displays the time and date the Query executed. For queries that execute more than once, the number of times that Query executed is displayed.
- For a SPUFI file, STROBE displays the time and date the SPUFI file was executed.
- If the Procedure was executed once, or if SQL=DETAIL=PROC was specified, STROBE displays the time and date the Procedure was executed.

Otherwise, STROBE displays the number of times the SQL statement was executed.

As shown in this report example, a DBRM named SQLDYNAM accounted for 46.53% of the CPU time. The next step is to examine the SQL statements within this DBRM.

CPU Usage by SQL Statement Report

The CPU Usage by SQL Statement report shows the distribution of CPU activity among SQL statements within a Procedure, Query, or DBRM and displays the target SQL statements that caused the activity.

There are two formats for the CPU Usage by SQL Statement reports: one for DBRMs, Queries, and SPUFI files (Figure 3-6 on page 3-10), and one for Procedures (Figure 3-19 on page 3-22). Both types include a Description section that displays the SQL statement text for all target statements, and a CPU usage section that details the amount of CPU activity consumed by each executing statement.

The Procedure reports display the structure of the Procedure, as well as information about the number of times each constituent Query, QMF command, and first-level Subprocedure was executed. (For a complete description of the CPU Usage reports for Procedures, see “Reporting QMF Procedure and Query Structures” on page 3-21.)

Figure 3-6. CPU Usage by SQL Statement Report for DBRMs

```
                                        ** CPU USAGE BY SQL STATEMENT **

  DBRM - SQLDYNAM                              CREATED - 23 JUN 99 16:12:38

    STATIC, NON-CURSOR SQL

            253 SELECT FIRSTNME INTO :H FROM VEMP WHERE FIRSTNME='LEO'
                                                                                                                PIO: YES

            258 SELECT COUNT(*) INTO :H FROM WPALXM.PIOTABLE
                                                                                                                PIO: YES

            262 SELECT COUNT(*),AVG(FIELD1) INTO :H,:H FROM WPAKRM.QAINTTB
                                                                                                                PIO: YES

    STATIC, CURSOR SQL

             01 DECLARE QAPIO CURSOR FOR SELECT COUNT(*),AVG(FIELD1) FROM WPAKRM.QAINTTB
                                                                                                                PIO: YES

             02 DECLARE EMP01 CURSOR FOR SELECT EMPNO FROM VEMP
             08 DECLARE DEPT1 CURSOR FOR SELECT DEPTNO,ADMRDEPT FROM VDEPT
    DYNAMIC, NON-CURSOR SQL

             07 UPDATE BOSTON01.DSN8310.VPHONE SET MIDDLEINITIAL=????                              LOCATION: BOSTON01
    DYNAMIC, CURSOR SQL

             03 SELECT EMPNO FROM DSN8310.VEMP
             04 SELECT DEPTNO FROM DSN8310.VDEPT
             05 SELECT ADMRDEPT FROM DSN8310.VDEPT
             06 SELECT DEPTNUMBER FROM DSN8310.VPHONE

    STMT   STATEMENT                 STMT-EXECUTION   % CPU TIME        CPU TIME HISTOGRAM  MARGIN OF ERROR:  1.28%
   NUMBER    TEXT                    CNT  AVG TIME     SOLO TOTAL        .00      4.50      9.00     13.50     18.00

      253 SELECT                      10    9.99        .19    .48        .+
      258 SELECT                      20   18.32       1.38   3.20        .***++++
      262 SELECT                       3    3.90       6.24  13.06        .*************+++++++++++++++
      282 OPEN         02 DECLARE      4    5.05        .00    .02        .
      308 OPEN         04 SELECT     134   66.45        .02    .02        .
      392 OPEN         05 SELECT       6    7.34        .02    .02        .
      334 FETCH        08 DECLARE     21   24,11        .10    .14        .
      343 FETCH        01 DECLARE      7    8.11       9.12  18.82        .*********************+++++++++++++++
      355 FETCH        02 DECLARE     12   14.56       2.95   3.85        .*******++
      366 FETCH        03 SELECT       8    9.21       4.31   5.05        .**********++
      404 FETCH        04 SELECT      13   14.23        .12    .15        .
      414 FETCH        05 SELECT      11   12.22        .29    .29        .
      285 CLOSE        02 DECLARE      5    6.23        .02    .02        .
      296 PREPARE      03 SELECT       3    4.31       1.09   1.35        .**+
      306 PREPARE      04 SELECT       9   10.66       1.26   1.52        .**+
      320 PREPARE      07 UPDATE       2    2.44       2.26   2.50        .*****+
                                     ---   -----      -----   ----
  DBRM - SQLDYNAM     TOTALS         258   54.32      29.37  46.53
  N
```

Report Header Description

If the CPU Usage by SQL Statement report is for a DBRM, Query, or SPUFI file, or if the user specified SQL=DETAIL=PROC when generating the Profile, STROBE displays a header showing the DBRM, Query, or SPUFI file name, and the following:

- for DBRMs, the creation date and time stamp
- for Queries that were executed once during the measurement session, or the user specified SQL=DETAIL=QUERY when generating the Profile, the execution time and date
- for SPUFI files, the execution time and date
- for Queries that were executed more than once during the measurement session, the number of times that Query was executed
- for Procedures that executed once, or if SQL=DETAIL=PROC was specified when the Profile was generated, STROBE displays the time and date the Procedure was executed.

Otherwise, STROBE displays the number of times the DBRM was executed.

SQL Statement Identification

SQL statements are listed under the following headings: “Static, Non-cursor SQL”, “Static, Cursor SQL”, “Dynamic, Non-cursor SQL”, or “Dynamic, Cursor SQL” . If STROBE encounters “Static, Non-cursor SQL” statements, each detail line begins with the precompiler listing statement number followed by statement type. The statement number generated for nonexecutable statements is not made available to DB2 during execution. Nonexecutable statements can only be embedded in an application program and include precompiler declaratives such as DECLARE CURSOR and WHENEVER.

In the CPU activity section of the report, STROBE begins detail lines with the precompiler statement number followed by the SQL verb. Under the heading STATEMENT TEXT, for SQL verbs with target SQL statements, STROBE displays the generated statement number and the SQL verb of the target statement. Statement numbers greater than 32k report as blank because of a DB2 restriction.

For nonexecutable SQL statements, STROBE shows a uniquely generated statement number. The statement number is based on the sequence in which activity is detected for a statement. This generated number is used to correlate the executable statement, for example OPEN or FETCH, with the target DECLARE CURSOR statement.

STROBE also replaces any arguments in dynamic statements with “????” and aggregates activity in each SQL statement that has the same row/column name and operand. For example, STROBE combines activity in the statements “SELECT name=William” and “SELECT name=Paul”, and reports it as “SELECT name=????”. For security, STROBE also replaces each character in a password or DSETPASS with a pound sign (#). Host variables may be replaced by “:H”.

If the report is for a Procedure, STROBE shows the structure of the Procedure, displaying any embedded Queries, QMF commands, and Subprocedures. The Procedure’s name precedes either a date and time or, if the Procedure was executed more than once and SQL=DETAIL=PROC was *not* specified when the Profile was generated, the number of times it was executed.

The example in Figure 3-6 on page 3-10 illustrates a report for a DBRM that uses both static and dynamic SQL statements. In the Query description, SQL statements are grouped by their type: “Static, Non-cursor SQL”, “Static, Cursor SQL”, “Dynamic, Non-cursor SQL”, or “Dynamic, Cursor SQL”). For more information on how to control the length of the text shown for a SQL statement, see “Creating the STROBE Performance Profile” on page 2-6.

LOCATION

The STROBE DB2 Feature offers limited support of remote SQL for DB2 Version 5 Release 1 and above. It reports location name for SQL statements querying against remote objects. The location name is indented on the line following the SQL text. If the SQL statement references objects at more than one remote location, the location name field displays “MANY”. For application programs using IBM’s Application Directed SQL, the STROBE DB2 Feature does not have access to full SQL statement text. Activity is reported using the precompiler statement number and SQL verb. SQL statements that reference both remote and local objects in different execution instances are labeled with the remote location name. All cursor operations issued under Application Directed SQL are categorized as “Static, Cursor SQL”. In this release, location information is provided for DBRMs and SPUFI queries only.

SQL statements executed remotely may show significant wait time. Occasionally, remote SQL will be reported without accompanying CPU activity. When this happens, refer to the Attribution Of CPU Wait Time report for percentages of wait attributed on behalf of the SQL. In addition, execution and wait time attribution reports for remote SQL may show attribution to VTAM interface module ISTAICIR on behalf of the SQL.

The CPU percentage shown in the CPU Usage By SQL Statement report for remote SQL does not include the remote site activity.

PARALLEL

If the DB2 Optimizer determines that an SQL statement meets the requirements for using parallel processing, STROBE displays a "PARALLEL" indicator for that SQL statement. This indicator appears on the line following the SQL text, and is followed by one or more fields that provide more specific information about the type of parallel processing observed. The PARALLEL indicator is not displayed for SQL statements that are not eligible. The following table lists the fields that appear with the PARALLEL indicator.

Field	Indicates whether or not the SQL statement
ELIG=YES|NO	was eligible for parallel processing.
PIO=YES|NO	used parallel I/O
CPU=YES|NO	was split and processed by multiple CPUs

STMT-EXECUTION CNT

In this column, STROBE displays the number of times the statement executed. This information only appears for DBRMs and SQL executed by the Remote Resource Services Attachment Facility (RRSAF). See "CPU Usage Reports for SQL" on page 3-7 for limitations on SQL statement count reporting.

STMT-EXECUTION AVG TIME

In this column, STROBE displays the average time the statement took to execute. This information only appears for DBRMs and SQL executed by the Remote Resource Services Attachment Facility (RRSAF).

Note: If you are using the Call Attachment Facility and a DSNTRACE DD statement is present in the job you are measuring, the statement count and execution average times reported by STROBE will be inaccurate.

CPU TIME SOLO

Percentage of time STROBE found only CPU servicing the SQL statement, with no concurrent I/O activity.

CPU TIME TOTAL

Percentage of time STROBE found the SQL statement engaged in activity with or without concurrent I/O activity.

Execution of Stored Procedures

When you measure a job that issues calls to DB2 (Version 5 Release 1 and above) stored procedures, the CPU Usage by SQL Statement reports (Figure 3-7 on page 3-13) shows the CALL statement executed, as well as CPU usage associated with its execution.

Figure 3-7. CPU Usage by SQL Statement Report for Stored Procedures

```
                                        ** CPU USAGE BY SQL STATEMENT **

DBRM - DB41STPC                               CREATED - 24 MAY 99 4:02:32

   STATIC, NON-CURSOR SQL

           89 CALL DB41STPD(:H,:H,:H,:H,:H,:H :H,:H :H,:H :H,:H :H,:H :H,:H :H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H
                 ,:H,:H,:H,:H,:H,:H)
          130 CALL DB41STPD('000140',:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H,:H
                 ,:H,:H)
 STMT   STATEMENT                                  CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  5.58%
 NUMBER    TEXT                                      SOLO     TOTAL        .00      6.50     13.00     19.50     26.00

     89 CALL                                         24.60     24.60        .************************************
    130 CALL                                         15.21     15.21        .***********************
                                                     -----     -----
 DBRM - DB41STPC                          TOTALS     39.81     39.81
```

The CPU time shown reflects the time utilized in executing the call statement. This time does not include CPU usage associated with the actual SQL statements contained and executed within the stored procedure. To gather CPU information associated with the execution of the stored procedure, you must measure the DB2 stored procedures address space (SPAS) concurrently with measuring the job that issues the CALL verb. The CPU usage shown for the stored procedure in the Performance Profile from the SPAS measurement may reflect multiple users invoking the same stored procedure.

Note: The STROBE Advanced Session Management Feature enables you to make requests to measure address spaces concurrently. When one measurement starts, you can designate other sessions to also begin. See STROBE *MVS User's Guide with Advanced Session Management* for details about making concurrent measurements.

Batch Transaction Reports

Transaction reports in the STROBE Performance Profile vary according to the type of address space and the STROBE features that were active during the measurement session. Transaction reports for TSO address spaces or batch processing programs show CPU activity by DB2 plan name (Figure 3-8 on page 3-14). When the STROBE DB2 Feature cannot attribute CPU activity in DB2 service modules to a DB2 plan name, it assigns that activity to the pseudo-transaction .DB2. When the STROBE DB2 Feature observes CPU activity in non-DB2 programs, it assigns that activity to the pseudo-transaction .NONDB2.

Note: STROBE does begin assigning non-DB2 CPU activity to .NONDB2 until it detects DB2 CPU activity. As a result, the total of non-DB2 CPU activity and DB2 CPU activity may not always equal 100 percent.

When you measure an online subsystem, the transaction reports show which transactions were responsible for most of the CPU activity. These reports are helpful when

- you want to see CPU usage for the transaction and CICS response time statistics
- the major CPU activity is *not* in DB2 service modules
- DBRMs are shared among transactions

For a description of transaction report examples for CICS measurements, see the section called "SQL CPU Usage Summary for SPUFI" on page 3-17.

Figure 3-8. Transaction Usage by Control Section Report

```
                                    ** TRANSACTION USAGE BY CONTROL SECTION **
TRANSACTION TESTBH31
 MODULE   SECTION                                        CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  2.61%
   NAME     NAME   COMPRESSED   FUNCTION                   SOLO     TOTAL        .00      4.00      8.00     12.00     16.00

  .DB2     DSNB1GET          RETRIEVE REQUESTED PAGE      .99       .99        .**
  .DB2     DSNB1REL          PAGE RELEASE ROUTINE         .28       .28        .
  .DB2     DSNB1SPF          SCH NORML PFTCH IN SEQ       .07       .07        .
  .DB2     DSNECP10          CNTL CSECT FOR DSNECP10     1.70      1.70        .****
  .DB2     DSNIDM   DSNILREP REPLAC POSBLY DISCON/CON     .14       .14        .
  .DB2     DSNIDM   DSNINXTP READ SPECFD PAG USNG MSA     .07       .07        .
  .DB2     DSNIDM   DSNIRNXT FETCH NEXT ROW TO PROG     12.23     12.23        .*******************************
  .DB2     DSNSLD1  DSNSFBK  GET/FREE FIX-LEN STORAGE     .07       .07        .
  .DB2     DSNSLD1  DSNSTACK STORAGE MGR STACK PROCSR    2.05      2.05        .*****
  .DB2     DSNSLD1  DSNSVSTK STACK STORAGE PROCESSOR     2.12      2.12        .*****
  .DB2     DSNTLM            IRLM SERVICE ROUTINES       .14       .14        .
  .DB2     DSNVGEPL          AGNT SERV/LATCH MGT DRVR    .07       .07        .
  .DB2     DSNVRMEL          RESOURC MGR LINKAGE CNTL    1.13      1.13        .**
  .DB2     DSNXEEZ  DSNXEEZ  DATA BASE PROGRM REQUEST    3.82      3.82        .*********
  .DB2     DSNXGRDS DSNXEBR  LINK CSECT RDS/ACCES MOD    2.90      2.90        .*******
  .DB2     DSNXGRDS DSNXECP  COPY APPLCTN STRUCTURES     6.58      6.58        .****************
  .DB2     DSNXGRDS DSNXERD  TOPMOST RDS CSECT          15.49     15.49        .**************************************
  .DB2     DSNXGRDS DSNXERT  APPLICATION CALL ROUTINE    7.43      7.43        .******************
  .DB2     DSNXGRDS DSNXRFN  RTIME FTCH NXT PROCESSNG    4.60      4.60        .***********
  .DB2     DSNXGRDS DSNXROUA RUN-TIME RET VALUES EXEC   10.25     10.25        .************************
  .DB2     DSN3EPX  DSNAPRHX DB2 SYSTEM SERVICES        12.80     12.80        .********************************
  .DB2     DSN3EPX  DSN3AA   DB2 SYSTEM SERVICES         2.26      2.26        .*****
  .IRLM    DXRRLM10          IRLM                         .42       .42        .*
  .IRLM    DXRRLM60          IRLM                         .57       .57        .*
  .NUCLEUS IEAVTSFR          SETFRR SERVICE              1.41      1.41        .***
  .NUCLEUS IEAVXSTK          S/B PCLINK STACK/UNSTACK    4.60      4.60        .***********
  .XMEMORY .XMEMORX          EXTENDED CROSS MEMORY       1.41      1.41        .***
  IRARMFIP                   SYSTEM RESOURCE MGMT        1.63      1.63        .****
  IRARMGLU                   SYSTEM RESOURCE MGMT         .14       .14        .
  TESTBC3  .DB2     DSNELI   TSO ATTACH LANG INTERFCE    1.77      1.77        .****
  TESTBC3  TESTBC3           .50       .50        .*
                                                        -----     -----
 TRANSACTION TESTBH31  TOTALS                          99.64     99.64
```

Reports for Analyzing Wait Time

The Performance Profile shows wait time in measurements of DB2 allied address spaces in the Wait Time by Module report and the Attribution of CPU Wait Time report. Two reports also show wait time detected in allied address spaces for SQL statements. The SQL Wait Summary and the Wait by SQL Statements reports use the same format as their equivalents for CPU usage. These reports are generated for DBRMs, Queries and Procedures.

Wait Time By Module Report

The Wait Time by Module report (Figure 3-9) shows the sites of wait time. In this TSO address space, the Wait Time by Module report shows wait time due to DB2 system module DSNVSR.

Figure 3-9. Wait Time by Module Report

```
                                        ** WAIT TIME BY MODULE **
MODULE   SECTION COMPRESSED   FUNCTION                RUN TIME PERCENT       RUN TIME HISTOGRAM  MARGIN OF ERROR:  2.53%
  NAME     NAME    SECTION                              PAGE     TOTAL        .00      1.50      3.00      4.50      6.00

 .DB2     DSNSLD1  DSNSVBK  GET/FREE VAR-LEN STORAGE     .00       .07        .
 .DB2     DSNVSR            SUSP/RES/CANCEL SYNCHRON     .00      4.01        .++++++++++++++++++++++++++
 .DB2     DSN3AMGP          DESC IEPLS SSAM GLB PAGE     .00       .07        .
                                                        -----     -----
 .DB2     TOTALS            DB2 SYSTEM SERVICES          .00      4.15
```

Attribution of CPU Wait Time Report

The Attribution of CPU Wait Time report (Figure 3-10) identifies the sites of invocation of system routines. When wait time is attributed to a DB2 service routine, the report header displays the invoked routine and shows the following:

- a transaction name, if one is available
- the SQL statement, DBRM, or Query name
- the time it was invoked or created
- its SQL statement number, if available
- the executed SQL statement's type and the target statement's number and type
- page wait and total wait time caused by the invoker

When wait time is attributed to DSNVSR, the report also displays the module and section name for the module that called DSNVSR, and provides a function descriptor for that module. The function descriptor can be helpful in identifying the resource for which the suspended SQL statement is waiting.

If you have identified a SQL statement on the Attribution of CPU Wait Time report that you want to examine in more detail, go the section "Wait By SQL Statement Report" on page 3-16.

Figure 3-10. Attribution of CPU Wait Time Report

```
                                        ** ATTRIBUTION OF CPU WAIT TIME **
.DB2     DSNVSR             SUSP/RES/CANCEL SYNCHRON
 --------------------------WAS INVOKED BY---------------------   -----------------VIA---------------------- WAIT TIME %
 XACTION  MODULE   SECTION    RETURN   LINE   PROCEDURE NAME       MODULE   SECTION  FUNCTION                 PAGE  TOTAL

 DSNESPRR                                                                                                    .00   .36
 DSNESPRR .DB2     DSNBBM             DB2 SYSTEM SERVICES                                                    .00   .05
 DSNESPRR .DB2     DSNVGEPL           AGNT SERV/LATCH MGT DRVR                                               .00 25.16

 XACTION  QUERY NAME                  TIME         STMT TEXT                  STMT TEXT

 DSNESPRR SIMT0001                    13:30:24          EXECUTE                 02 DROP DATABASE             .00   .01
                                                   VIA MODULE DSNVGEPL           AGNT SERV/LATCH MGT DRVR
 DSNESPRR SIMT0001                    13:30:24          EXECUTE                 04 COMMIT                    .00  2.12
                                                   VIA MODULE DSNVGEPL           AGNT SERV/LATCH MGT DRVR
 DSNESPRR SIMT0001                    13:30:24          EXECUTE                 06 CREATE TABLESPACE         .00  1.20
                                                   VIA MODULE DSNVGEPL           AGNT SERV/LATCH MGT DRVR
 DSNESPRR SIMT0001                    13:30:24          EXECUTE                 07 CREATE TABLE              .00   .09
                                                   VIA MODULE DSNVGEPL           AGNT SERV/LATCH MGT DRVR
  .
  .
  .
 DSNESPRR SIMT0001                    13:30:24          EXECUTE                 08 CREATE INDEX              .00   .56
                                                   VIA MODULE DSNVGEPL           AGNT SERV/LATCH MGT DRVR
 DSNESPRR SIMT0004                    13:37:00          EXECUTE                 03 INSERT                    .00   .61
                                                   VIA MODULE DSNVGEPL           AGNT SERV/LATCH MGT DRVR
                                                                                                          ----- -----
                                                                                                          .00 45.17
```

SQL Wait Time Summary Report

The SQL Wait Summary report (Figure 3-11 on page 3-16) shows the distribution of wait time among the Queries, Procedures, and DBRMs that invoked DB2 system services. Each detail line in this report shows the SQL type, its name, the time and date of execution (or, for DBRMs, the time and date it was created), and its page and total run time. Also for DBRMs, STROBE provides the total number of executed statements and a weighted average for the amount of time it took for all statements to execute. The report fields are described in the section called "SQL CPU Usage Summary Report" on page 3-7.

Figure 3-11. SQL Wait Summary Report

```
                                          ** SQL WAIT SUMMARY **
 SQL    SQL              STMT-
EXECUTION    TIME/COUNT          % RUN TIME        CPU TIME HISTOGRAM  MARGIN OF ERROR:  1.28%
 TYPE   NAME              CNT  AVG TIME                      PAGE    TOTAL       .00     12.50     25.00     37.50     50.00

 DBRM  SQLDYNAM            24   1.5267   16:13:36 23JUN99    .03     .91      *
 DBRM  SQLDYNM1            26    .8164   16:13:06 23JUN99    .00     .45      *
 DBRM  SQLDYNM2          5069    .0166   16:12:38 23JUN99    .00     .52      *
                                                            -----   -----
       TOTAL WAIT ACTIVITY                                   .03    1.89
```

Wait By SQL Statement Report

The Wait by SQL Statement report attributes wait time to individual statements. The example in Figure 3-12 on page 3-17 shows how STROBE presents this information. Refer to "CPU Usage by SQL Statement Report" on page 3-9 for a description of the report fields other than RUN TIME PERCENT, which are described next.

RUN TIME PERCENT

RUN TIME PERCENT is the percentage of time during the measurement session that the address space was in wait state. There are two measures of run time:

- PAGE shows wait time that results from retrieving a page from the page data set. A high value in this column indicates that there is not enough physical memory assigned to the address space. If you noticed a high page rate on the Measurement Session Data report, the Wait by SQL Statement report shows you which statement was experiencing delay because of paging.
- TOTAL measures all causes of wait time, including page retrieval, programmed I/O operations, and timer requests.

Figure 3-12. Wait by SQL Statement Report for DBRMs

```
                                        ** WAIT BY SQL STATEMENT **

DBRM - DB31PRA4                              CREATED - 23 JUN 99 16:12:38

  STATIC, NON-CURSOR SQL

          253 SELECT FIRSTNME INTO :H FROM VEMP WHERE FIRSTNME='LEO'
                                                                                                                PIO: YES

          258 SELECT COUNT(*) INTO :H FROM WPALXM.PIOTABLE
                                                                                                                PIO: YES

          262 SELECT COUNT(*),AVG(FIELD1) INTO :H,:H FROM WPAKRM.QAINTTB
                                                                                                                PIO: YES

  STATIC, CURSOR SQL

           01 DECLARE QAPIO CURSOR FOR SELECT COUNT(*),AVG(FIELD1) FROM WPAKRM.QAINTTB
                                                                                                                PIO: YES

           02 DECLARE EMP01 CURSOR FOR SELECT EMPNO FROM VEMP
           08 DECLARE DEPT1 CURSOR FOR SELECT DEPTNO,ADMRDEPT FROM VDEPT
  DYNAMIC, NON-CURSOR SQL

           07 UPDATE BOSTON01.DSN8310.VPHONE SET MIDDLEINITIAL=????                              LOCATION: BOSTON01
  DYNAMIC, CURSOR SQL

           03 SELECT EMPNO FROM DSN8310.VEMP
           04 SELECT DEPTNO FROM DSN8310.VDEPT
           05 SELECT ADMRDEPT FROM DSN8310.VDEPT
           06 SELECT DEPTNUMBER FROM DSN8310.VPHONE

 STMT   STATEMENT                 STMT-EXECUTION   % RUN TIME    RUN TIME HISTOGRAM  MARGIN OF ERROR:  1.28%
NUMBER    TEXT                   CNT  AVG TIME     SOLO TOTAL    .00      4.50      9.00     13.50     18.00

   253 SELECT                      9     .99       .99    .99  .
   258 SELECT                     20   22.32     10.60    .06  .*************
   262 SELECT                      4    3.90       .46    .46  .
   282 OPEN          02 DECLARE    2    1.05       .23    .23  .
   308 FETCH         04 DECLARE  134  145.12     22.12  29.03  .*******************************++++++++
   392 FETCH         05 DECLARE    2    4.11       .23    .23  .**
   334 EXEC IMMEDIATE 08 EXPLAIN   1    2.11       .23    .23  .
   343 FETCH         01 SELECT     1    2.11       .23    .23  .
   366 PREPARE       03 SELECT     8   36.21       .92   1.38  .*
   404 PREPARE       04 SELECT     2     .11       .23    .46  .
   414 PREPARE       05 SELECT     5     .11       .46    .46  .
   285 PREPARE       02 SELECT     7    3.19       .69    .92  .
                                 ----  ------      ----   ----
                                 195  117.56     36.40  45.68
```

Reports Customized for QMF and SPUFI

This section contains reports customized for QMF and SPUFI.

SQL CPU Usage Summary for SPUFI

The sample report in Figure 3-13 on page 3-18 shows activity in several Queries submitted through SPUFI. STROBE displays each execution of SPUFI input file QRYMEM01 as separate detail lines.

Figure 3-13. SQL CPU Usage Summary Report for a SPUFI File

```
                                    ** SQL CPU USAGE SUMMARY **

 SQL     SQL                          TIME/COUNT          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR - 1.25%
 TYPE    NAME                                              SOLO     TOTAL        .00      8.00     12.00     18.00     24.00

 SPUF    QRYFIL1                 14:20:04  16MAY99        17.82     18.54        .*******************************+++
 SPUF    QRYMEM01                14:10:24  16MAY99        29.24     29.24        .************************************
 SPUF    QRYMEM01                14:15:51  16MAY99         8.82      8.82        .***********
                                                          -----     -----
       TOTAL SQL CPU USAGE                                47.88     48.60
```

CPU Usage by SQL Statement Reports for SPUFI

The CPU Usage by SQL Statement report for SPUFI files shows the distribution of CPU activity among SQL statements within the file and displays the SQL statements that caused the activity. Figure 3-14 shows an example of a report where STROBE identifies the file name as QRYMEM01. The text of the target SQL statement contained in the file is then displayed, as well as the CPU times for the statements that were executed as a result of that target statement. For example, SPUFI used the executing OPEN, FETCH, and PREPARE statements to process the target SELECT statement. STROBE does not display a statement number in the STMT NUMBER column for SPUFI.

Figure 3-14. CPU Usage by SQL Statement Report for SPUFI Files

```
                                    ** CPU USAGE BY SQL STATEMENT **

SPUFI FILE - QRYMEM01                                    EXECUTED -08 JUL 99 14:10:24

           01  SELECT * FROM DSN8110.EMP
           02  UPDATE VEMPLOYEE SET PHONENUMBER = ???? WHERE EMPLOYEENUMBER = ????
           03  GRANT BSDS,RECOVER TO PARKER,LUTZ

 STMT   STATEMENT                                  CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR - 1.98%
NUMBER   TEXT                                      SOLO     TOTAL        .00     2.00      4.00      6.00      8.00

       OPEN                  01 SELECT             1.08      1.08        .****
       FETCH                 01 SELECT             8.54      8.54        .*****************************************
       PREPARE               01 SELECT             2.72      2.72        .************
       PREPARE               02 UPDATE             5.82      5.82        .*************************
       PREPARE               03 GRANT              1.26      1.26        .******
       EXECUTE               02 UPDATE             4.37      4.37        .******************
       EXECUTE               03 GRANT              5.45      5.45        .***********************
                                                   -----     -----
QUERY QRYMEM01                          TOTALS     29.24     29.24
```

SQL CPU Usage Summary Reports for QMF

The STROBE DB2 Feature is able to identify QMF Procedures and QMF commands within a Procedure by tracing QMF SQL statements. When STROBE initializes a measurement session for a QMF application, it inserts the QMF Trace Attributor, a front-end intercept to the QMF Governor Exit (an installation-modified part of QMF). The Trace Attributor records the start and end of Procedures, QMF commands, and Queries. Although STROBE can identify the active Query (and the current SQL statement) without the intercept, it needs the intercept to identify:

- whether (in some cases) the active Query is a continuation or a re-execution of a previous, identical Query
- the name of the active Procedure
- SQL statements that are never executed (for example, DECLARE) and other SQL statements, Queries, and QMF Commands that are never “active” during a measurement session
- QMF Commands

The QMF Trace Attributor is active only for address spaces that STROBE measures; other QMF users are not affected. The Trace Attributor, like the QMF Governor Exit, runs under the same task as QMF.

Note: If you invoke the STROBE DB2 Feature after your QMF Procedure has begun executing, STROBE cannot accurately determine the structure of the Procedure and will report the activity as if the Queries were executed independently. STROBE will collect Procedure-related information, however, for any subsequent Procedure that it detects.

Identifying Unnamed QMF Procedures and Queries

When you DISPLAY a Procedure online and then issue the RUN command, or when you execute a Query or Procedure through batch with the IMPORT and RUN commands, the SQL statements that execute are unnamed, even if the original Query or Procedure was named. STROBE attributes activity in such unnamed Queries and Procedures to the pseudo-name .PROC*xxxx* or .QUERY*xxxx,* respectively, where *xxxx* is a four-digit number generated by STROBE. STROBE begins with 0001 and increments the number by 1 for each unnamed Procedure and Query encountered during the measurement session. For example, STROBE would attribute activity in the sixth unnamed Procedure to .PROC0006 and activity in the sixth unnamed Query to .QUERY0006.

The example in Figure 3-15 summarizes activity in four types of Queries submitted through QMF: a Saved Query, a Prompted Query, a Query by Example, and an interactive SQL statement. In this example, two executions of SARWW.DBFETCH are reported in aggregate. Note that STROBE places the number of executions in the TIME/COUNT column.

Figure 3-15. SQL CPU Usage Summary Report

```
                                    ** SQL CPU USAGE SUMMARY **

SQL     SQL                          TIME/COUNT          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR - 1.25%
TYPE    NAME                                              SOLO     TOTAL        .00      8.00     12.00     18.00     24.00

QSAV    SARWW.DBFETCH                  2                  21.39     21.39        .**********************************
PRMQ                                14:50:19  16MAY99     18.54     18.54        .******************************
QBE                                 14:45:11  16MAY99      8.82      8.82        .************
SQL                                 14:30:06  16MAY99     11.11     11.11        .****************
PROC PUBACCTINFOPROCOO              15:00:15  16MAY99     20.27     20.27        .**********************************
                                                          -----     -----
      TOTAL SQL CPU USAGE                                 80.13     80.13
```

STROBE automatically aggregates all executions of a Query or Procedure into one line in this report and indicates this aggregation by showing the number of executions (rather than the time of execution) in the TIME/COUNT column. STROBE always aggregates activity for DBRMs and SPUFI. You can suppress this aggregation for Queries and Procedures by specifying SQL=DETAIL=(PROC,QUERY) when you produce the Performance Profile. (For more information on suppressing aggregation, see "Attribution Reporting" on page 2-9.)

Figure 3-16 on page 3-20 shows an SQL CPU Usage Summary report for QMF Procedures where STROBE used its front-end intercept into the QMF Governor Exit and the SQL=DETAIL=PROC option was set. This example shows three types of Queries and one Procedure.

Figure 3-16. SQL CPU Usage Summary Report for QMF Procedures

```
                                        ** SQL CPU USAGE SUMMARY **

QUERY   QUERY                        TIME/COUNT          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR - 1.25%
TYPE    NAME                                              SOLO     TOTAL        .00      8.00     12.00     18.00     24.00

QSAV    SARWW.DBFETCH           14:40:02  16AUG99          8.37      8.37        .********
QSAV    SARWW.DBFETCH           14:50:02  16AUG99         13.02     13.02        .******************
PRMQ                            14:50:19  16AUG99         17.82     18.54        .*****************************++
QBE                             14:45:11  16AUG99          8.82      8.82        .*********
PROC    PUBACCTINFOPROC00       15:00:15  16AUG99         20.27     20.27        .***********************************
                                                          -----     -----
                                                          80.13     80.13
        TOTAL CPU USAGE BY QUERIES                        55.28     56.00
```

Figure 3-17 shows the SQL CPU Usage Summary report when the user specifies SQL=NOPROC or when STROBE does *not* insert its front-end intercept into the QMF Governor Exit. In this report, the Procedure PUBACCTINFOPROC00 is replaced by the Queries and commands that it contains, and total CPU usage for the Procedure is not shown.

Figure 3-17. SQL CPU Usage Summary Report for QMF Procedures Using SQL=NOPROC

```
                                        ** SQL CPU USAGE SUMMARY **

SQL     SQL                          TIME/COUNT          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR - 1.25%
TYPE    NAME                                              SOLO     TOTAL        .00      8.00     12.00     18.00     24.00

QSAV    SARWW.DBFETCH           14:40:02  14MAY99          8.37      8.37        .**********
QSAV    SARWW.DBFETCH           14:50:02  14MAY99         13.02     13.02        .**********************
QSAV    SARWW.DBQUERY10         15:11:13  13MAY99          4.01      4.01        .****
QSAV    SARWW.DBQUERY10         15:13:00  13MAY99          4.45      4.45        .*****
QSAV    SARWW.DBSEL             15:00:32  16MAY99          4.45      4.45        .*****
QSAV    SARWW.DBUPDAT           15:03:42  12MAY99          4.80      4.80        .******
PRMQ                            14:50:19  16MAY99         17.82     18.54        .*********************************
QBE                             14:45:11  16MAY99          8.82      8.82        .*************
SQL                             14:30:56  16MAY99         11.11     11.11        .*******************
                                                          -----     -----
      TOTAL SQL CPU USAGE                                 80.13     80.13
```

CPU Usage by SQL Statement Reports for Saved Queries

The CPU Usage by SQL Statement report for Queries shows the distribution of CPU activity among SQL statements within the Query and displays the SQL statements that caused the activity. Figure 3-18 on page 3-20 shows the CPU Usage by SQL Statement report for a Saved Query. Because such a Query can contain only one SQL statement, all the executing statements listed in the CPU section of the report were executed on behalf of that target statement. STROBE does not display a statement number in the STMT NUMBER column for QMF queries.

Figure 3-18. CPU Usage by SQL Statement Report for a Saved Query

```
                                        ** CPU USAGE BY SQL STATEMENT **

QUERY - SARWW.DBFETCH                                   EXECUTED - 08 AUG 99 14:35:18

           01  SELECT * FROM DSN8110.EMP

 STMT   STATEMENT                                    CPU TIME   PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR - 1.98%
NUMBER    TEXT                                        SOLO     TOTAL        .00     3.00      6.00      9.00     12.00

          OPEN                 01 SELECT              1.42      1.42        .*****
          FETCH                01 SELECT             11.13     11.13        .**********************************
          PREPARE              01 SELECT              3.51      3.51        .************
          EXECUTE              01 SELECT              5.33      5.33        .****************
                                                     -----     -----
QUERY SARWW.DBFETCH                         TOTALS   21.39     21.39
```

Reporting QMF Procedure and Query Structures

Procedures can invoke Queries, other Procedures (henceforth called Subprocedures), and QMF commands. When STROBE displays the structure of a Procedure in a report, it shows the order in which each of these elements is invoked and, in the case of Subprocedures, the elements each invokes. When Subprocedures are nested within other Subprocedures, the path can be difficult to follow. Every Subprocedure expands until only SQL statements, Queries, and QMF commands appear. STROBE marks the path by including the nesting level in front of each statement's type.

If, for example, the code for the main Procedure L1P05 and its Subprocedures L2P03, L2P04, and L3P01 is

Procedure L1P05 (main Procedure)

```
IMPORT PROC L2P03 FROM 'SA.STDD.QMF.IMPORT'(MEMBER=L2P03 CONFIRM=NO)
RUN PROC L2P03
RUN PROC L2P04

Procedure L2P03 (1st-level Subprocedure)
RUN QUERY SRCHQRY
RUN QUERY SRCHQRY

Procedure L2P04 (1st-level Subprocedure)
RUN PROC L3P01

Procedure L3P01 (2nd-level Subprocedure)
RUN QUERY NXTNAME
RUN PROC L2P03
```

STROBE displays Procedure L1P05's structure as shown in Figure 3-19 on page 3-22.

Activity in Procedures

The CPU Usage by SQL Statement report for Procedures (Figure 3-19 on page 3-22) shows how STROBE reports the structure of the Procedure as well as the distribution of CPU activity among the SQL statements it invoked. STROBE follows this list with a breakdown of the CPU activity used by each Query and command invoked by the Procedure. (To change the way the STROBE DB2 Feature reports Procedures, see "Separating Procedures and Queries into Individual Entities" on page 2-7.)

As shown in the previous example, the Procedure L1P05 contains one command and four embedded Procedures. The beginning and end of each Procedure is shown by the PROC and PROC-END statements within the description. For example, Procedure L2P03 executes as a first-level Procedure, then it is called by Procedure L3P01. STROBE shows the initial execution on lines 2-5 and the second execution on lines 9-12.

STROBE indicates each statement's nesting level by placing a two-digit number in front of the statement type. For example, both the IMPORT command and Subprocedure L2P03 are first-level items, so their names are preceded by "01". Subprocedure L2P04 invokes Subprocedure L3P01, which, in turn, invokes Subprocedure L2P03. STROBE denotes this structure by placing "01" in front of PROC L2P04, "02" in front of PROC L3P01, and "03" in front of PROC L2P03.

Figure 3-19. CPU Usage by SQL Statement Report for Procedures

```
                                        ** CPU USAGE BY SQL STATEMENT **

 PROC - L1PO5                                EXECUTED - 08 MAY 99 19:44:41

          01  CMD         IMPORT
        01 PROC        L2PO3
          02  QRY         SRCHQRY
          02  QRY         SRCHQRY
        01 PROC-END    L2PO3
        01 PROC        L2PO4
          02 PROC      L3PO1
          03  QRY         NXTNAME
          03 PROC      L2PO3
          04  QRY         SRCHQRY
          04  QRY         SRCHQRY
          03 PROC-END  L2PO3
          02 PROC-END  L3PO1
        01 PROC-END    L2PO4

 COMMAND - IMPORT                            NO SQL ACTIVITY

              IMPORT PROC L2PO6 FROM 'SASEH.QMF.IMPORT'(MEMBER=L2PO6 CONFIRM=NO

 SUBPROC L2PO3

 QUERY - SRCHQRY                             EXECUTED    2 TIMES

            01 SELECT NAME05,NAME04,NAME03,NAME02,NAME01 FROM TBL00001

 STMT        STATEMENT                          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  4.10%
NUMBER           TEXT                             SOLO     TOTAL        .00       .50      1.00      1.50      2.00

   134       PREPARE                01 SELECT              1.93      1.93        .**************************************
                                                          -----     -----
 QUERY - SRCHQRY                             TOTALS       1.93      1.93

 END SUBPROC L2PO3

 SUBPROC L2PO4

 QUERY - NXTNAME                             EXECUTED - 08 MAY 99 19:44:43

            01 SELECT NAME02,NAME01 FROM TBL00001

 STMT        STATEMENT                          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  4.10%
NUMBER           TEXT                             SOLO     TOTAL        .00       .50      1.00      1.50      2.00

   438        PREPARE                01 SELECT               .70       .70        .**************
                                                          -----     -----
 QUERY - NXTNAME                           TOTALS         .70       .70

 QUERY - SRCHQRY                             EXECUTED    2 TIMES

            01 SELECT NAME05,NAME04,NAME03,NAME02,NAME01 FROM TBL00001

 STMT        STATEMENT                          CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  4.10%
NUMBER           TEXT                             SOLO     TOTAL        .00      1.00      2.00      3.00      4.00

  237      PREPARE                01 SELECT              2.46      2.46        .************************
                                                        ------     -----
 QUERY - SRCHQRY                           TOTALS         2.46      2.46

 END SUBPROC L2PO4
                                                        ------     -----
 PROC - L1PO5                              TOTALS         5.09      5.09
```

Subprocedure Identification

If the Procedure invokes a first-level Subprocedure, STROBE precedes the CPU information for the Queries and commands contained in that Subprocedure with the line "SUBPROC *procname*" and follows those reports with the line "END SUBPROC *procname*." (STROBE denotes any Subprocedures below first-level when it displays the Procedure's structure.)

Query Identification

For each Query or QMF command that the Procedure invokes, STROBE displays a header showing the statement's type, its name, and one of the following:

- if *both* the Query and Procedure were executed once, the date and time the Query was executed

- if *either* the Query or the Procedure was executed more than once, the number of times the Query was executed during a single run of the Procedure
- if the QMF command does not cause SQL activity, the statement "NO SQL ACTIVITY"

CPU Usage by Queries and Commands

STROBE follows the header for each active Query and command header with the distribution of CPU activity for that statement.

Chapter 4.
The STROBE Performance Profile for DB2 Services Address Spaces

The DB2 services address spaces that STROBE measures are

- database services address space (DBSAS)
- system services address space (SSAS)
- stored procedures address space (SPAS)

Interpreting the STROBE Performance Profile for a DB2 DBSAS

DB2 database activity and a limited amount of CPU activity occur in a DB2 database services address space (DBSAS). The DB2 service modules residing in the DBSAS are most often executed in cross-memory mode by the allied address spaces, so most of the CPU activity in these modules is shown in the STROBE Performance Profile for the allied address spaces.

In the Performance Profile for the DB2 DBSAS, STROBE reports show the I/O facility usage associated with the VSAM data sets that contain the DB2 databases and that are processed by the MVS Media Manager.

A DB2 database comprises a collection of DB2 tables and occupies one or more table spaces. Each DB2 table space is backed by one or more VSAM entry sequenced data sets (ESDSs), whose names contain a node identifying the DB2 table spaces they back. When (as is typically the case) only one DB2 table occupies a table space, time spent accessing the VSAM ESDSs backing that table space actually represents time spent accessing a specific DB2 table.

When the STROBE DB2 Feature measures a DB2 DBSAS, the Performance Profile for that address space shows the following:

- DB2 database I/O facility usage by device, volume, data definition name, and cylinder
- data set characteristics
- CPU activity in tasks in the DB2 DBSAS
- function descriptors for the DB2 modules

Measurement Session Data Report

The Measurement Session Data report describes the environment during a measurement session. The low CPS time percent in this example (Figure 4-1 on page 4-2) is typical of a database address space because most DB2 CPU activity is performed by the allied address spaces accessing DB2 databases.

In the SUBSYSTEM field, the “characters” before the release level of DB2 indicate what type of address space that was measured. For example "SS" is the abbreviation for Systems Services address space. "DB" indicates a DB2 DBSAS. "SP" indicates a Stored Procedures address space.

Figure 4-1. Measurement Session Data Report

```
                                        ** MEASUREMENT SESSION DATA **

-------- JOB ENVIRONMENT --------        ------- MEASUREMENT PARAMETERS ------        ----- MEASUREMENT STATISTICS ----

 PROGRAM MEASURED   -    DSNYASCP        ESTIMATED SESSION TIME  -      20 MIN        CPS TIME PERCENT -           0.05
 JOB NAME           -    WPA1DBM1        TARGET SAMPLE SIZE      -      10,000        WAIT TIME PERCENT -         99.95
 JOB NUMBER         -    STC30816        REQUEST NUMBER   (A)    -        150        RUN MARGIN OF ERROR PCT  -    .98
 STEP NAME   -           WPA1DBM1        FINAL SESSION ACTION    -       QUIT        CPU MARGIN OF ERROR PCT  -    .83

 DATE OF SESSION    -    09/01/01        SYS REQ -  SCS01                           TOTAL SAMPLES TAKEN -      10,000
 TIME OF SESSION    -    08.30.57                                                   TOTAL SAMPLES PROCESSED -  10,000
                                                                                   INITIAL SAMPLING RATE-   8.33/SEC
                                        --------- REPORT PARAMETERS ---------      FINAL SAMPLING RATE  -   8.33/SEC
 SYSTEM -     ESA SP4.3.0
 DFSMS              -       1.1.0       REPORT RESOLUTION    -       32 BYTES      SESSION TIME -   20 MIN 48.22 SEC
 SUBSYSTEM   -       DB2 DB 3.1.0       SORTSIZE             -        999,999      CPU TIME -        0 MIN  0.68 SEC
 CPU MODEL          -   3090-600S       LINES/PAGE           -             60      WAIT TIME    -   20 MIN  5.81 SEC
 SMF/SYSTEM ID -       SESA/SCS01                                                  STRETCH TIME -    0 MIN 41.73 SEC
                                        DASD= 2.0%  DASDGAP= 5  ATTR= 0.0%
 REGION SIZE BELOW 16M  -  5,064K       SQL=MAXLEN=9999                           SRB TIME     -    0 MIN 18.67 SEC
 REGION SIZE ABOVE  -     32,768K                                                 SERVICE UNITS-               657

 PTF LVL- 2.4.0.FS000000/FS000000                                                  PAGES IN-      0     OUT-       0
 STROBE TAPE NUMBER -  000-S00DSK                                                  PAGING RATE  -           0.00/SEC
                                                                                   EXCPS -          2       0.00/SEC

 SAMPLE DATA SET  - ZZ.WPA1DBM1.S001D001
```

Time and Resource Demand Distribution Reports

The Time Distribution of Activity Level report shows the distribution over time of CPU activity and data set use during the measurement session. The report in Figure 4-2, shows activity in order of decreasing solo time, indicating that the data set identified by the system-generated data definition name SYS00012 is used heavily throughout the measurement session. The sums in the individual columns in the report are seldom greater than 10, indicating little overlap among data set access and CPU activities.

Figure 4-2. Time Distribution of Activity Level Report

```
                                ** TIME DISTRIBUTION OF ACTIVITY LEVEL **


TASK OR   RESOURCE  N X 10 PLUS OR MINUS 5 IS PERCENT OF FULL UTILIZATION    * IS GREATER THAN 95%     - IS LESS THAN 5%
 DDNAME           .-------------------------------------------------------------------------------------------------------.
DSNVEUS3  CPU     .-                            -- -  - 1                          -                      -          .
DSNVEUS31 CPU     .-                                     - ---1 -  --- -                 -                    -     .
DSNYASCP  CPU     .                                                                                                  .
SYS00012  3380    .76884786299*8867748886874554634563112333646674445453535245645 63---1---1--11311-2 -  -1 1-   - -1  - .
SYS02273  3380    .                                        -111322111121212141 677851        -     -111412332221      .
SYS00044  3380    .           -   -                  12    31-2433   -14-                    -                    1  .
SYS00010  3380    .         --   -                 -- - -- -- -- - --- -------- -1-   -1  1  ---1--   1 -- 12----  12 --.
SYS00006  3380    .         -    -       --1-11-1----   -- --11---- 1----- ------    -- 1--    2-   -1   1111-  - 2- 1-.
SYS02305  3380    .                                311-1                  -    243       -    1    -               .
SYS00008  3380    .          -   -         - -     -              -  -     - --    -  -             -31               .
SYS02573  3380    .                                34                                                              .
SYS00238  3380    .         -                     --- ----- -- 1----- ---    - ----  - -1    1     -     1- -   - -  - .
    OTHER         .1- -2 1121--   1111   1-1 - 111-1-1-1111--211423111114311-11342121336-11-1-1-1-111---11---- 311--11-.
                  0----0----1----1----2----2----3----3----4----4----5----5----6----6----7----7----8----8----9----9----*.
                  0----5----0----5----0----5----0----5----0----5----0----5----0----5----0----5----0----5----0----5----*.
                  START RUN                PERCENT OF ALLOCATED RUN TIME                                  END RUN
```

The Resource Demand Distribution report (Figure 4-3 on page 4-3) lists, in decreasing order of solo time, the usage associated with highly active data sets. This report indicates that the data set identified as SYS00012 is active for nearly 35 percent of the measurement session.

Figure 4-3. Resource Demand Distribution Report

```
                                   ** RESOURCE DEMAND DISTRIBUTION **

     ACTIVITY          ---- PERCENT OF RUN TIME -----      -------- PERCENT OF RUN TIME SPENT -------- CUMULATIVE PERCENTAGES
TASK OR   RESOURCE    SERVICED    SERVICED    SERVICED       SOLO       SOLO       SOLO      CAUSING      SOLO      CAUSING
 DDNAME               BY CPU      BY I/O    BY EITHER      IN CPU     IN I/O    IN EITHER   CPU WAIT      TIME    CPU WAIT

DSNVEUS3  CPU            .34         .00         .34          .10        .00         .10       00.00         .10       00.00
DSNVEUS31 CPU            .20         .00         .20          .09        .00         .09       03.92         .09       03.92
DSNYASCP  CPU            .00         .00         .00          .00        .00         .00       99.46         .00       99.46
SYS00012  3380           .00       34.59       34.59          .00      27.43       27.43       00.00       27.62       99.46
SYS02273  3380           .00        8.59        8.59          .00       4.81        4.81       00.00       32.43       99.46
SYS00044  3380           .00        2.51        2.51          .00       1.81        1.81       00.00       34.24       99.46
SYS00010  3380           .00        2.08        2.08          .00       1.23        1.23       00.00       35.47       99.46
SYS00006  3380           .00        2.22        2.22          .00       1.00        1.00       00.00       36.47       99.46
SYS02305  3380           .00        1.71        1.71          .00        .76         .76       00.00       37.23       99.46
SYS00008  3380           .00         .60         .60          .00        .43         .43       00.00       37.66       99.46
SYS02573  3380           .00         .72         .72          .00        .39         .39       00.00       38.05       99.46
SYS00238  3380           .00         .88         .88          .00        .33         .33       00.00       38.38       99.46
  OTHER   3380           .36        8.98        9.34          .21       3.70        3.91       00.00       42.29       99.46
```

Data Set Characteristics Report

The Data Set Characteristics report (Figure 4-4) provides the data set names associated with the data definition names shown in the reports above. By noting the identifying qualifier for the table space in the name of the heavily used data set, you can turn to the I/O Facility Utilization Summary report shown in Figure 4-6 on page 4-4 to identify the DB2 table space with activity causing heavy DASD activity.

Figure 4-4. Data Set Characteristics Report

```
                                   ** DATA SET CHARACTERISTICS **

DDNAME     ACCESS   POOL    REC   BLK/CI  HBUF BUF    RPL  - SPLITS-      EXCP  DATA SET NAME
           METHOD     NO    SIZE   SIZE     NO  NO   STRNO    CI   CA     COUNTS

SYS0005  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB06.SYSUSER.I0001.A001
SYS0006  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB06.SYSPLAN.I0001.A001
SYS0010  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB06.SCT02.I0001.A001
SYS0012  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB06.SYSDBASE.I0001.A001
SYS0016  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB06.SYSVIEWS.I0001.A001
SYS0044  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB06.SYSLGRNG.I0001.A001
SYS0493  MEDIA MGR                                                        DB2T.DSNDBD.DSNDB07.DSNTMP01.I0001.A001
SYS2273  MEDIA MGR                                                        TEST.DSNDBD.PADBT86.PATST860.I0001.A001
```

Data Set Characteristics Supplement Report

The Data Set Characteristics Supplement report (Figure 4-5 on page 4-4) provides additional information on data sets. It can be cross-referenced to the Data Set Characteristics report by data definition name, access method, and data set name. Because DB2 data sets are processed by Media Manager, only the ddname, access method, and data set name are available to STROBE.

Compuware recommends that you suppress the Data Set Characteristics Supplement report when measuring DB2 address spaces. STROBE reports all data set names and the report can become very large when all data sets used by Media Manager are included. For more information on the Data Set Characteristics Supplement report, see Chapter 3 of *STROBE MVS Concepts and Facilities*.

Figure 4-5. Data Set Characteristics Supplement Report

```
                                    ** DATA SET CHARACTERISTICS SUPPLEMENT **

SYS0005 MEDIA MGR DB2T.DSNDBD.DSNDB06.SYSUSER.I0001.A001

SYS0006 MEDIA MGR DB2T.DSNDBD.DSNDB06.SYSPLAN.I0001.A001

SYS0010 MEDIA MGR  DB2T.DSNDBD.DSNDB06.SCT02.I0001.A001

SYS0012 MEDIA MGR DB2T.DSNDBD.DSNDB06.SYSDBASE.I0001.A001

SYS0016 MEDIA MGR  DB2T.DSNDBD.DSNDB06.SYSVIEWS.I0001.A001

SYS0044 MEDIA MGR  DB2T.DSNDBD.DSNDB06.SYSLGRNG.I0001.A001

SYS0493 MEDIA MGR DB2T.DSNDBD.DSNDB07.DSNTMP01.I0001.A001

SYS2273 MEDIA MGR TEST.DSNDBD.PADBT86.PATST860.i0001.A0001
```

I/O Facility Utilization Summary Report

The I/O Facility Utilization Summary report summarizes I/O access by device, volume, and ddname. The example in Figure 4-6 shows that two of the most active data sets, SYS00012 and SYS00044, reside on the same unit and volume. Moving these data sets to separate volumes may decrease their access times and thus improve the response times in the online address spaces associated with this DBSAS.

Figure 4-6. I/O Facility Utilization Summary Report

```
                                  ** I/O FACILITY UTILIZATION SUMMARY **


 UNIT        DEVICE       VOLUME     DDNAME          RUN TIME PERCENT       RUN TIME HISTOGRAM  MARGIN OF ERROR - 0.55%
  NO          TYPE         ID                          SOLO     TOTAL       .00      9.00     18.00     27.00     36.00

 134     DA 3380          DB2T93    SYS00005           .02       .03        .
 134     DA 3380          DB2T93    SYS00012         27.43     34.59        .**********************************++++++++
 134     DA 3380          DB2P93    SYS00044          1.81      2.51        .**
                                                      -----     -----
UNIT  134 TOTALS                                     29.26     37.13

 168     DA 3380          DB2T03    SYS00006          1.00      2.22        .*+
 168     DA 3380          DB2T03    SYS00010          1.23      2.08        .*+
 168     DA 3380          DB2T03    SYS00016           .25       .76        .
                                                      -----     -----
UNIT  168 TOTALS                                      2.48      5.06

 BE2     DA 3380          TEST66    SYS02273          4.81      8.59        .*****++++
```

Reports That Show CPU Activity

Because CPU usage in this example is extremely low (the Measurement Session Data report in Figure 4-1 on page 4-2 shows only 0.05 %), trying to improve use of CPU time in this case is probably not worthwhile. The CPU reports can, however, serve as a baseline. If subsequent measurement sessions show much higher CPU activity, the causes of higher activity should be determined and evaluated.

Program Section Usage Summary Report

The Program Section Usage Summary report (Figure 4-7 on page 4-5) shows the distribution of CPU time used by each active control section of each active module in the DB2 address space. DB2 and other IBM system modules are compressed and shown as pseudo-sections under the pseudo-module .SYSTEM.

Figure 4-7. Program Section Usage Summary Report

```
                                    ** PROGRAM SECTION USAGE SUMMARY **


 MODULE  SECTION   16M   SECT  FUNCTION                CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  5.38%
   NAME     NAME   <,>   SIZE                            SOLO     TOTAL       .00     12.00     24.00     36.00     48.00

.SYSTEM  .COMMON               COMMON AREA               2.11      2.41        .*+
.SYSTEM  .DB2                  DB2 SYSTEM SERVICES      34.34     34.94        .****************************+
.SYSTEM  .MEDIAMG              MEDIA MANAGER MODULES      .60       .60        .
.SYSTEM  .NUCLEUS              MVS NUCLEUS                .30       .30        .
.SYSTEM  .SVC                  SUPERVISOR CONTROL       46.99     47.59        .*************************************
.SYSTEM  .VSAM                 VIRTUAL STORAGE ACC METH 12.95     12.95        .**********
                                                        -----     -----
.SYSTEM  TOTALS                SYSTEM SERVICES          97.29     98.79

IEFQB551             >   11184 SWA MANAGEMENT             .60       .60        .
IGC0013I             >  105184 SUPERVISOR SERVICES        .60       .60        .
                                                        -----     -----
PROGRAM DSNYASCP    TOTALS                              98.49     100.00
```

Program Usage by Procedure Report

The Program Usage by Procedure report (Figure 4-7) shows, for DB2 and other system routines, the modules and control sections responsible for the CPU activity in the DB2 address space. Function descriptors appear for each control section or, if STROBE has not gathered control section information, for each module.

Figure 4-8. Program Usage by Procedure Report

```
                                    ** PROGRAM USAGE BY PROCEDURE **


         .SYSTEM   SYSTEM SERVICES                        .DB2      DB2 SYSTEM SERVICES

 MODULE   SECTION        FUNCTION            INTERVAL  CPU TIME PERCENT       CPU TIME HISTOGRAM  MARGIN OF ERROR:  5.38%
   NAME      NAME                             LENGTH   SOLO     TOTAL       .00      1.00      2.00      3.00      4.00

 DSNB1CPP            CLOSE PAGESET PIECE         7592    .60       .60        .******
 DSNB1CPS            CLOSE PAGE SET              6552    .30       .30        .***
 DSNB1GET DSNB1GET   RETRIEVE REQUESTED PAGE     12000   3.01      3.01        .********************************
 DSNB1OPS            OPEN PAGESET               11872    .30       .30        .***
 DSNB1OST            OPEN PAGESET PIECE SERVC    7648    .60       .60        .******
 DSNB1REL            PAGE RELEASE ROUTINE        4336    .90       .90        .*********
 DSNB5FOR            PREFORMAT DATASET PAGES     3376    .30       .30        .***
 DSNB5RDP DSNB5RDP   READ DATABASE PAGE ROUTINE  3568   2.11      2.11        .**********************
 DSNIDM   DSNICUBD   DATA MANAGEMENT DRIVER      1640    .30       .30        .***
 DSNIDM   DSNIDSTT   DISPLAY DBD/PAGESET INFO    2840   2.11      2.11        .**********************
 DSNIDM   DSNIFOD    FMT HEADR/SPACE MAP PAGS    4512    .30       .30        .***
 .
 .
 .
 DSNIDM   DSNITCUS   MSI  SET CUB STAT=BEFORE    5392    .30       .30        .***
 DSNIDM   DSNITDLE   DEL POSBLY DISCON RECORD   10848    .60       .60        .******
 DSNKDM   DSNKCLOK   LOCK INDEX PAGE CONDITION   3312    .90       .90        .*********
 DSNKDM   DSNKFIND   SEARCH INDEX FOR KEY        9120   1.50      1.50        .***************
 DSNKDM   DSNKLTCH   OBTAIN INDEX TREE LOTCH     2480    .30       .30        .***
 DSNKDM   DSNKSLFV   INTERNAL PROCESSING         1512    .60       .60        .******
 DSNKDM   DSNKULTH   RELEASE INDEX TREE LOTCH    2632    .90       .90        .*********
 DSNPDSM  DSNPAEDL   ADD DROP LIST ENTRY          992    .30       .30        .***
 DSNPDSM  DSNPDEDL   DELETE THE DROP LIST         848    .30       .30        .***
 DSNPDSM  DSNPDPSO   DROP PAGESET SPACE          2904    .90      1.20        .*********+++
 DSNPDSM  DSNPGNPO   GET NUMBR OF PCS TO OPEN    2576    .60       .60        .******
 DSNPDSM  DSNPSCTO   SHOW CATALOG                3784    .60       .60        .******
 DSNSLD1  DSNSVBK    GET/FREE VAR-LEN STORAGE    6176   1.20      1.51        .************+++
 DSNSLD1  DSNSVSTK   STACK STORAGE PROCESSOR     3344    .90       .90        .*********
 DSNTLM              IRLM SERVICE ROUTINES      21856   1.20      1.20        .************
 DSNVEUS3            SERVICE TASK DISPATCHER    10376    .60       .60        .******
 DSNVSR   DSNVSR     SUSP/RES/CANCEL SYNCHRON    5312   2.41      2.41        .************************
 DSNXKAP  DSNXKAL    CATALOG ACCESS LOCATOR     15480   1.81      1.81        .******************
 DSNXKAP  DSNXKBLM   BUILD MSI FOR READ OPERS   10704   1.81      1.81        .******************
                                                        -----     -----
         .DB2       TOTALS                             34.26     34.87
```

STROBE Performance Profile for Stored Procedures Address Spaces

DB2 implements an additional DB2 services address space called the stored procedures address space (SPAS). DB2 uses the SPAS to execute DB2 stored procedures on behalf of calling programs. When the STROBE DB2 Feature measures this address space, it produces a standard Performance Profile that shows SQL and CPU usage by stored procedures. For more information, see “Execution of Stored Procedures” on page 3-12.

Index

Special Characters

A

B

C

D

E

I

M

N

P

Q

R

S

T

U

V

W