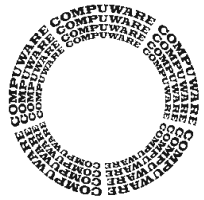
STROBE MVS

STROBE UNIX System Services 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. CWSTUU3A
January 23, 2003.

Contents

Figures

Summary of Changes

This section lists the changes to the Unix System Services Feature for STROBE MVS for Sysplex Release 3.0.

Changes to the STROBE Unix System Services Feature

The Unix System Services Feature for STROBE MVS for Sysplex Release 3.0 provides support for Performance Profile reporting when an measured application is using a file with a z/FS structure and also now supports measurement of forked processes using ADD QUEUED requests.

Changes to this Manual

The Unix System Services Feature for STROBE MVS for Sysplex Release 3.0 provides examples of the Data Set Characteristics Report that include z/FS file data.

Introduction

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

The STROBE Application Performance Measurement System and the STROBE *UNIX System Services Feature* are products designed for IBM MVS/ESA, IBM OS/390, and IBM z/OS systems. The STROBE *UNIX System Services Feature* is designed for use with IBM OS/390 (V2 release 4 and higher), program number 5647-A01 and z/OS V1 Release 1 program number 5694-A01.

How This Manual Is Organized

Chapter 1, "Overview" presents an overview of the STROBE *UNIX System Services Feature* and the OS/390 UNIX System Services environment.

Chapter 2, "Using the STROBE UNIX System Services Feature" describes how to use STROBE with the STROBE *UNIX System Services Feature* to measure an application that uses OS/390 UNIX System Services.

Chapter 3, "Analyze the Performance Profile" explains how to interpret the STROBE Performance Profile reports from the measurement of an application executing in an OS/390 UNIX System Services address space.

How to Use This Manual

You should read Chapter 1, "Overview" and Chapter 2, "Using the STROBE UNIX System Services Feature" before submitting a measurement request. To interpret a Performance Profile for an application executing in an OS/390 UNIX System Services address space, read Chapter 3, "Analyze the Performance Profile".

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 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 Management System is a product that determines how and where application time is spent in online and batch-processing environments. STROBE tracks activity within the OS/390 environment by collecting several types of data and producing reports that determine how to improve an application's performance.

The STROBE UNIX System Services Feature enables customers to measure batch-processing and online applications that use OS/390 UNIX System Services. This chapter describes the benefits of the STROBE UNIX System Services Feature, outlines concepts and terminology of the STROBE product, and provides an overview of the OS/390 UNIX System Services environment.

Benefits of the STROBE UNIX System Services Feature

The STROBE UNIX System Services Feature obtains application performance information as the application executes. STROBE organizes this information into a series of reports called the STROBE Performance Profile. The Performance Profile identifies:

- CPU activity and wait time for modules running in OS/390 address spaces that call OS/390 UNIX System Services
- the use of the Hierarchical File System (HFS)
- the use of the z/FS file system
- how applications use UNIX System Services

The STROBE UNIX System Services Feature helps you to measure, analyze, and improve the performance of applications that use OS/390 UNIX System Services, enabling you to develop and maintain more efficient and responsive applications throughout the application life cycle.

The next section discusses the key concepts and terms that are central to the use of the STROBE UNIX System Services Feature.

Concepts and Terminology

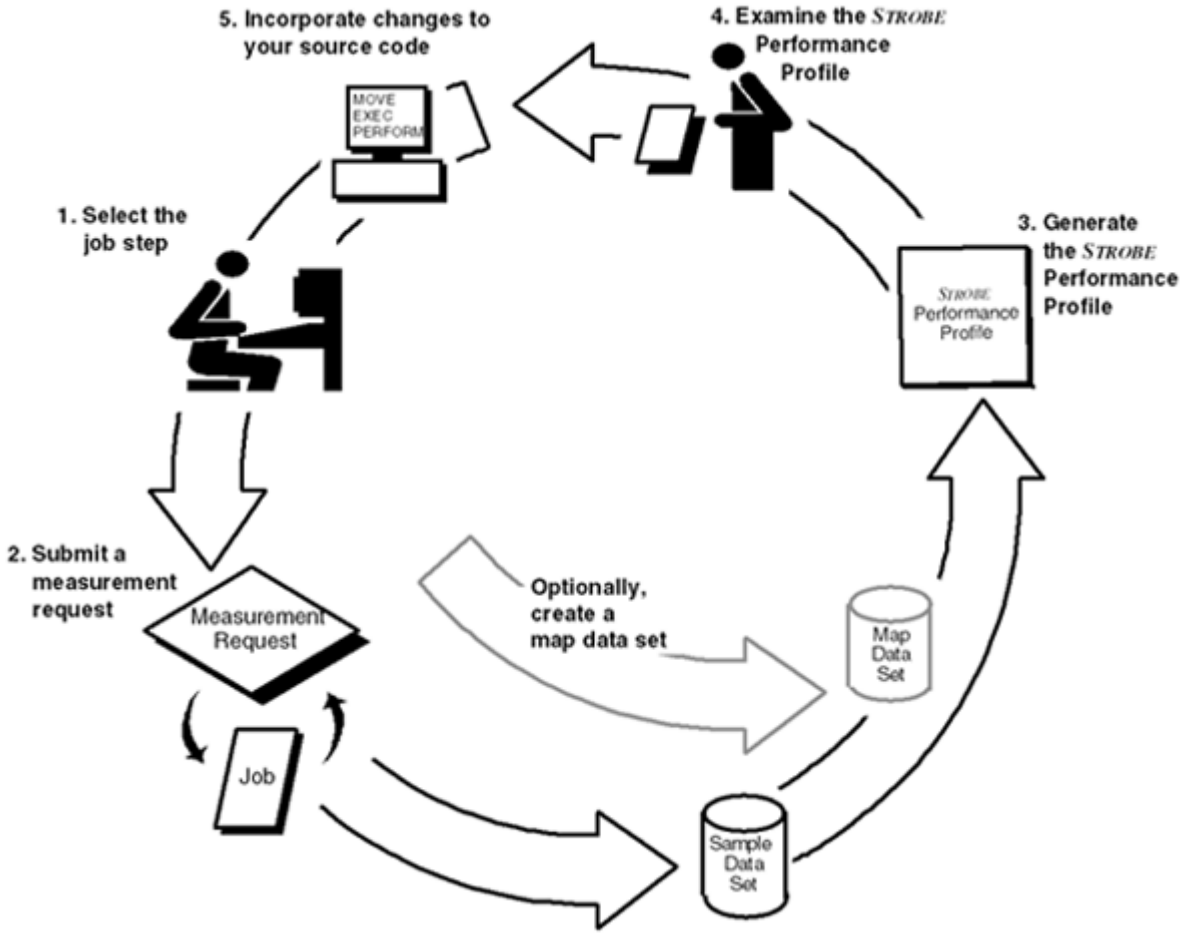
Before using the STROBE UNIX System Services Feature, it is helpful to familiarize yourself with the basic concepts and terminology specific to STROBE. As shown in Figure 1-1 on page 1-2, you select the job step you want to measure and submit a measurement request.

A measurement request specifies the parameters for measuring the performance of an application. When the application is active, STROBE begins a measurement session, an interval where it collects performance data about the application while it is executing. STROBE stores measurement data in a sample data set, a file that contains the information collected during a single measurement session. Each measurement session corresponds to one sample data set.

After STROBE closes the sample data set, you can use it to create the Performance Profile, a hierarchical series of reports that presents the performance data collected during a

measurement session. These reports show where and how the application spends its time during execution. Examine the Performance Profile to identify performance improvement opportunities, and incorporate the appropriate changes to your source code. You can then repeat the process to see the results of your code changes, and identify other potential performance improvement opportunities that may have been masked by the initial problem.

Figure 1-1. Overview of STROBE Tasks



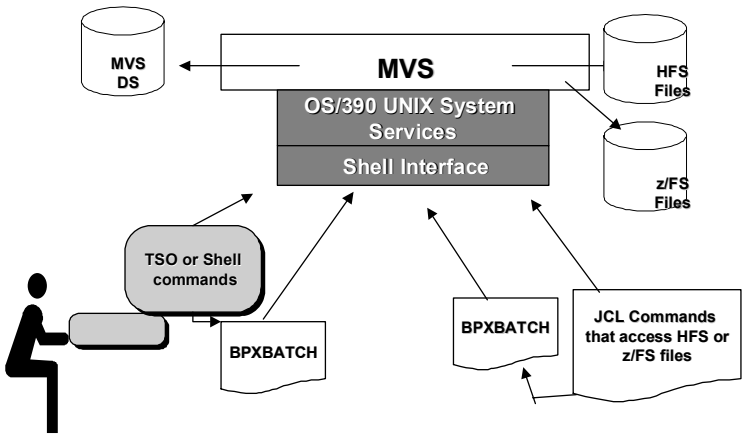
Overview of the OS/390 UNIX System Services Environment

The OS/390 UNIX System Services environment comprises a shell interface, a kernel address space, and either the Hierarchical File System (HFS) or the z/FS file system required by the z/OS version of the operating system. Each component of this environment enables you to execute a number of OS/390 UNIX System Services application services. The application services include support for

- utilities to administer and develop applications in the UNIX System Services environment
- socket applications
- direct telnet service based on the TCP/IP protocol
- full-screen applications

Each component of the OS/390 UNIX System Services environment is briefly described in the following sections.

Figure 1-2. Overview of the OS/390 UNIX System Services Environment



The Interactive Shell Interface

The interactive shell interface is an execution environment that processes requests for online or batch processing applications and OS/390 services.

Customers access the shell interface, use the Time Sharing Option Extension (TSO/E), the BPXBATCH utility, the rlogin command, or the application itself to issue shell commands or invoke utilities, as shown in Figure 1-2. Once accessed, the shell interface interprets and executes the shell commands within a shell script. A shell script, which is a file of your shell commands that becomes executable when you specify a file name, creates a parent process in an address space to execute the program.

Once parent processes are created in an address space, they may create additional forked or spawned child processes to carry out program commands. The fork command creates a process in a separate address space. The spawn command creates a process in the same address as the parent (local spawn) or in its own address space (non-local spawn).

The Kernel Address Space

The kernel address space acts as an interface between OS/390 and the hardware and provides services for system calls, file support, and device drivers. It contains the MVS support for OS/390 UNIX System Services, including programs for I/O, management, and communication.

An OS/390 program or a job invoked from a TSO/E session requests kernel services by using C/C++ functions, shell commands (after invoking the shell), or callable services. Compuware recommends that you do not measure the kernel address space.

The Hierarchical File System

OS/390 UNIX System Services stores its data and program files into a Hierarchical File System (HFS). The files are organized in a hierarchy and stored in directories.

HFS includes data and executable files. HFS and the traditional MVS file system are compatible; you use TSO/E shell commands to copy, store, and link files between the HFS file system and MVS data sets. In addition, you can use the OS/390 UNIX System Services Network File System (NFS) Feature to issue commands from your OS/390 UNIX System Services workstation to read or write data from an MVS data set.

The NFS Feature mounts HFS files, data sets, or directories into an empty directory at your workstation and links them to an MVS data set. After you mount or link files, you can edit, run, or write to another file from your workstation. This feature allows you to access remote files as though they were local to your workstation.

Customers access HFS files by entering shell commands on the TSO/E command line or submit JCL statements or BPXBATCH that specify the name of the file in the HFS directory

The z/FS File System

z/FS is a new file system which is used in conjuction with the HFS file system in support of the z/OS bimodal operating system. HFS files are used as a base upon which the zFS files are mounted. From an application point of view, z/FS files look and act the same as HFS files. The HFS and z/FS file systems are mostly equivalent, except that z/FS supports space sharing between z/FS file systems. HFS file systems are not capable of sharing space. z/FS files also are mounted and customers can access them using the same methods as HFS files.

Where to Find More Information

You can find detailed instructions for submitting and managing measurement requests and creating Performance Profiles in the STROBE/ISPF Online Tutorial and the *STROBE MVS User's Guide* or the *STROBE MVS User's Guide with Advanced Session Management*. For more information on interpreting Performance Profiles, see *STROBE MVS Concepts and Facilities.*

Chapter 2.
Using the STROBE UNIX System Services Feature

The STROBE UNIX System Services Feature enables you to measure applications that call OS/390 UNIX System Services.You can find detailed instructions for submitting measurement requests in the STROBE/ISPF Online Tutorials, the *STROBE MVS User's Guide*, and the *STROBE MVS User's Guide with Advanced Session Management*.

There are a few considerations to keep in mind when submitting measurement requests in an OS/390 UNIX System Services environment. This chapter discusses these considerations, and describes how to create a STROBE Performance Profile.

Measuring the Application

With STROBE, you can measure applications that run as batch jobs, use TSO, or execute in online regions. This section describes how to measure applications executing in the OS/390 UNIX System Services environment. The STROBE UNIX System Services Feature collects measurement data at the address space level for any processes running in the measured address space. STROBE reports on the CPU usage and wait times for the address space in the form of the Performance Profile.

Measuring an Active Job

To measure an active job, complete the following steps:

1. Select Option 1 (ADD ACTIVE) from the STROBE OPTIONS menu. The STROBE - ADD ACTIVE REQUEST panel (Figure 2-1) appears.

Figure 2-1. The STROBE - ADD ACTIVE REQUEST Panel

```
------------------------- STROBE - ADD ACTIVE REQUEST -------------------------

JOBNAME ===>             (Jobname or clear to list active jobs/processes)
                          Enter J for Jobs, P for UNIX Processes ===> P

SYSTEM  ===>             (System or clear to list available systems)
SCHEDULE REQUEST ===> N  (Y or N; Use Y only when overriding the default)
AUTO PROFILE CREATION  ===> N (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   ===> FAC      (Notify when session completes)

SAMPLE DATA SET INFORMATION:
  DATA SET NAME PREFIX ===> ZZ
  UNIT NAME   ===> 3380     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      SAVE REQUEST IN GROUP        ===> N
```

2. Specify the target system in the SYSTEM field or clear this field to select from a list of systems.

3. Perform one of the following steps:
   - Specify the target job step in the JOBNAME field. If you are submitting shell commands or shell scripts from TSO, enter the TSO ID.
   - Clear the JOBNAME field and enter a "J" to view all active jobs or a "P" to view all active processes. Press **Enter**.

   Note: To display processes, you must be using OS/390 Version 2 Release 6 or higher.

   If you entered a "J": the STROBE - ACTIVE JOB SELECTION LIST panel appears. This panel lists all active jobs on the system. Enter an "S" to the left of the job to select it for measurement. Press **Enter**.

   If you entered a "P": the STROBE - ACTIVE PROCESS SELECTION LIST panel appears (Figure 2-2). The following fixed fields provide additional information for the process:
   - PID, the process identification number, assigned by OS/390
   - LVL, the level in the process hierarchy in which the process resides
   - PGPID, the process group identification number
   - USER, the creator of processes listed in the JOBNAME field

   Note: To add or change the order of fields appearing to the right of the USER field, refer to the STROBE - ACTIVE PROCESS DISPLAY FORMAT panel. See "Changing the Format of the STROBE - ACTIVE PROCESS SELECTION LIST Panel" on page 2-3 for more information.

   Enter an "S" to the left of the process to select it for measurement. Press **Enter**.

Figure 2-2. The STROBE - ACTIVE PROCESS SELECTION LIST Panel

```
                -- STROBE - ACTIVE PROCESS SELECTION LIST --

S - Select job to be measured (Press ENTER to update list)    SYSTEM: SYSA
Change Display Format  ===>

S    PID      LVL    PGPID     USER   | COMMAND                  JOBNAME  ASID
  ========== === ========== ======== | ======================== ======== ====
           1   0          1 TCPIP    | BPXPINPR                 BPXOINIT 0018
    16777218   1   16777218 TCPIP    | BPXVCLNY                 NFSC     001B
    16777219   1   16777219 TCPIP    | GFSCMAIN                 NFSC     001B
           4   1          4 TCPIP    | EZBTCPIP                 TCPIP    0032
           5   1          5 TCPIP    | EZACFALG                 TCPIP    0032
           6   1          6 BPXOINIT | EZASASUB                 TCPIP    0032
   117440519   1  117440519 BPXOINIT | GFSCRPCD                 NFSC     001B
           8   1          8 FTPD     | FTPD                     FTPD1    0037
           9   1          9 TCPIP    | EZBTMCTL                 TCPIP    0032
          10   1         10 BPXOINIT | EZBTTMST                 TCPIP    0032
    33554443   1   33554443 BPXOINIT | BPXVCMT                  NFSC     001B
          12   1         12 INETD    | /usr/sbin/inetd /etc/ine INETD8   003A
  1509949466   2 1509949466 INETD    | otelnetd -Y 206.33.85.16 INETD2   0059
  1040187420   3 1040187420 INETD    | sh -L                    INETD2   0059
          13   1         13 NFSC     | GFSCRPCD                 NFSC     001B
          14   1         14 NFSC     | GFSCRPCD                 NFSC     001B
```

4. Configure the measurement session by specifying the following information on the STROBE - ADD ACTIVE REQUEST panel (Figure 2-1 on page 2-1) :
   - the estimated run time in the SESSION DURATION field
   - how many measurement samples you want STROBE to take in the TARGET SAMPLE SIZE field

   – the TSO user ID in the TSO USER ID TO NOTIFY field. STROBE notifies this user ID when the measurement request is complete.

5. Specify the sample data set information in the SAMPLE DATA SET INFORMATION fields.
6. Specify any additional measurement options by entering "Y" in the corresponding field and then pressing **Enter**. You can:
   – specify special data collectors
   – specify additional module mapping facilities
   – add session management parameters
   – request a retention time frame

STROBE then displays the selected panels. When you press **Enter** on the last panel, STROBE submits the measurement request. See the STROBE *MVS User's Guide* or the *STROBE MVS User's Guide with Advanced Session Management* for more information on these measurement options.

Changing the Format of the STROBE - ACTIVE PROCESS SELECTION LIST Panel

To change the order of fields or add additional fields appearing on the STROBE - ACTIVE PROCESS SELECTION LIST panel, enter a "Y" in the Change Display Format field. The STROBE - ACTIVE PROCESS DISPLAY FORMAT panel appears. Put the fields in order by numbering them in the Display Order column. The PID, LVL, PGPID, and USER fields are fixed and always appear as the first four fields of the STROBE - ACTIVE PROCESS SELECTION LIST panel, as shown in Figure 2-2 on page 2-2.

Measuring a Job That Is Not Yet Executing

To add a measurement request for a job step that is not yet executing, select Option 2 (ADD QUEUED) from the STROBE OPTIONS menu.

Complete the following information in the STROBE - ADD QUEUED REQUEST panel (Figure 2-3 on page 2-4):

1. Specify the name of the target job in the JOBNAME field.

   Note: For STROBE to correctly identify activity in the program invoking BPXBATCH, you must apply the appropriate PTF for IBM APAR OW38618. Contact IBM to obtain the PTF that is compatible with the specific release of your operating system. This PTF causes all spawns in the BPXBATCH address space to be local spawns, but only when STROBE is present in that address space. However, STROBE will not be able to measure a BPXBATCH login shell which requires a remote spawn.

2. Identify the target job in the PROGRAM or STEP field according to what you want to measure. To specify a job by its program name, enter the name as it appears in the PGM= operand of the EXEC statement.
3. Specify the target system in the SYSTEM field or clear this field to select from a list of systems.
4. Configure the measurement session by specifying:
   – the estimated run time in the SESSION DURATION field
   – how many measurement samples you want STROBE to take in the TARGET SAMPLE SIZE field
   – the TSO user ID in the TSO USER ID TO NOTIFY field. STROBE notifies this user ID when the measurement request is complete.

5. Specify the sample data set information in the SAMPLE DATA SET INFORMATION fields.
6. Specify any additional measurement options by entering "Y" in the corresponding field and then pressing **Enter**. You can:
   - specify special data collectors
   - specify additional module mapping facilities
   - add session management parameters
   - request a retention time frame

STROBE displays the selected panels. When you press **Enter** on the last panel, STROBE submits the measurement request. See the STROBE *MVS User's Guide* or the *STROBE MVS User's Guide with Advanced Session Management* for more information on these measurement options.

Figure 2-3. The STROBE - ADD QUEUED REQUEST Panel

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

JOBNAME   ===> WPAFXC    (Jobname)
  PROGRAM ===> WPAFXC    (Program)
  STEP    ===>                  (Name, number or step.procstep)
SYSTEM    ===> SCS01     (System or clear for a list or *ALL for all systems)

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

MEASUREMENT SESSION INFORMATION:
  SESSION DURATION     ===> 5         (Estimated time in minutes)
  TARGET SAMPLE SIZE   ===> 20000     (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
```

Creating the Performance Profile

Once you have the information you need (the sample data set and, optionally, the map data sets), you can create a Performance Profile for the application.

You have three options for creating a Performance Profile with STROBE/ISPF:

- Create a Performance Profile automatically when submitting a measurement request for an active or queued request.

  Enter "Y" in the AUTO PROFILE CREATION field on the STROBE - ADD ACTIVE REQUEST panel or the STROBE - ADD QUEUED REQUEST panel.

- Create a Performance Profile from the STROBE OPTIONS menu.

  Select Option 4 (PROFILE) from the STROBE OPTIONS menu. STROBE displays the STROBE - PRODUCE A PERFORMANCE PROFILE panel, where you can specify options for the Performance Profile.

- Create a Performance Profile from the STROBE - STATUS panel.

Enter "P" to the left of a completed measurement request number on the STROBE - STATUS panel. STROBE displays the STROBE - PRODUCE A PERFORMANCE PROFILE panel, where you can specify options for the Performance Profile.

For more information about creating a Performance Profile and codeblock detailed reporting, see the *STROBE MVS User's Guide* or the *STROBE MVS User's Guide with Advanced Session Management*.

Controlling the Level of Detail in the Performance Profile

By specifying certain STROBE options, you can either compress or expand the level of detail of the reports in the Performance Profile.

You can control the size of the reports by specifying certain parameters when you create the Performance Profile. For example, on a number of reports, you can specify a minimum percentage of activity that must occur for a procedure or cylinder to appear as an individual line on the report. If the minimum percentage is not attained, contiguous procedures or cylinders are condensed on a single line in the report. The lower the number that you specify, the greater the level of detail in the reports.

The following reports have minimum percentage options:

- Program Usage by Procedure
- DASD Usage by Cylinder
- Transaction Usage by Control Section
- Attribution

You can also break the reports into manageable sections by specifying the report resolution; that is, the number of bytes to be considered a codeblock for detailed reporting. Once again, the lower the number you specify, the greater the level of detail in the reports. For more information, see *STROBE MVS Concepts and Facilities* or *STROBE MVS User's Guide* or *STROBE MVS User's Guide with Advanced Session Management*.

Creating a Profile That Relates CPU Activity with Source Statement Text

For C/C++ applications that call OS/390 UNIX System Services, the STROBE UNIX System Services Feature can associate CPU activity to specific statement numbers. The process of creating this association is called indexing. Associating the reported statement number with the source code text is a four-step process. You must:

- create a compiler listing data set
- create a map data set from the compiler listing data set
- edit the map data set
- create the Performance Profile, providing the map data set as input

Since you do not need this level of detail for all modules, first create a Performance Profile without providing any map data sets as input. Then review the Performance Profile to determine the modules for which you want to see statement-specific information. You need to create map data sets only for the modules in which you are interested.

The following sections describe how to create a C/C++ compiler listing data set with the appropriate compile options, and how to create the map data set. For detailed information on this process, see the *STROBE MVS User's Guide* or the *STROBE MVS User's Guide with Advanced Session Management*.

Create the Compiler Listing Data Set

If you are interested in viewing C/C++ statement and line number information, you must provide a compiler listing data set with the appropriate compile options.

To prepare a C/C++ compiler listing data set, first compile the C/C++ application. Set the environmental variable to **CXX_STEPS = 0Xffffffff** to force the use of the Prelinker option, as shown in the following example:

```
  set -o allexport
_CXX_STEPS=0xffffffff
c++  \
   -v  \
   -W 0,list,source,csect  \
   -W 1,p,map  \
   -W 1,map,list  \
   myprog.C  \
   1>myprog.list \
   2>myprog.err  \
   &
```

Next, convert the compiler listing data set to an MVS sequential file using the following DCB characteristics, **VBA, LRECL=137.** Once you create the compiler listing data set, you are ready to create the map data set, as described in the following sections.

Create the Map Data Set

After you have a compiler listing data set, your next step is to create a map data set. A map data set is a repository for the information required to relate addresses in the object code with statements in the source program.

To create a map data set, select Option 5 (INDEX) from the STROBE OPTIONS menu and press **Enter**. STROBE displays the STROBE-INDEX TO CREATE A MAP DATA SET panel.

To complete the STROBE indexing procedure, specify the following:

- "Y" in the C language field
- the name of the map data set in the OUTPUT: MAP DATA SET field
- the name of the compiler listing data set in one of the INPUT: COMPILER SYSPRINT DATA SETS fields

Edit the Map Data Set

Edit the map data set only if a control section's name is a longname or greater than eight characters. For all longnames, STROBE truncates these names to the first eight characters when creating the map data set. STROBE will not provide indexed information in the Performance Profile for the control section because it does not recognize the truncated name. To create an indexed Performance Profile, replace the truncated name with the STROBE-generated token name.

Note: A *token* is an eight-character STROBE-generated identifier. The token comprises the first four characters of a module or section name followed by a hyphen (-) and then the last three characters of the name.

To edit the map data set:

1. Determine the token STROBE uses to represent the truncated name by referring to the Token-Longname Cross Reference report (Figure 3-6 on page 3-6).
2. For each longname, overwrite the truncated name in the map data set with the token from the Token-Longname Cross Reference report.

Create the indexed Performance Profile as described in Chapter 3 of the *STROBE MVS User's Guide* or Chapter 2 of the *STROBE MVS User's Guide with Advanced Session Management.*

Chapter 3.
Analyze the Performance Profile

Once your measurement request is complete and you have created the STROBE Performance Profile, you are ready to analyze and identify performance improvement opportunities from the STROBE Performance Profile.

This chapter is a guide for analyzing the STROBE Performance Profile for an application executing in an OS/390 UNIX System Services address space. A complete description of the STROBE Performance Profile is found in Chapter 3 of *STROBE MVS Concepts and Facilities*.

About the Application

The example in this chapter refers to a Performance Profile produced for an application that had been recently ported to a UNIX System Services environment running in OS/390. The application uses information from a variety of investment files to calculate the present value of a customer's investment portfolio and create a report detailing the investments. The application was originally written and executed in a client/server UNIX environment and was used by a single customer representative in a small branch office.

The application was ported to the OS/390 environment so that a customer service representative could run the program. However, they soon detected performance problems in the application, which prompted the system administrator to use STROBE to analyze the application and look for performance improvement opportunities.

Verifying the Measurement Session Information

The starting point for analyzing any Performance Profile is the Measurement Session Data report (Figure 3-1 on page 3-2). This report describes the computing environment during a measurement session and provides measurement statistics for the measurement session.

There are several report fields that help you determine if the Performance Profile is valid and which reports in the Performance Profile you should analyze.

To verify the validity of the Performance Profile, examine the following fields:

- In the JOB ENVIRONMENT column, ensure that the PROGRAM MEASURED, DATE OF SESSION, and TIME OF SESSION match your measurement request.
- In the MEASUREMENT PARAMETERS column, ensure that the appropriate version of OS/390 UNIX System Services appears in the SUBSYSTEM field.

  Note: Though OS/390 UNIX System Services are technically part of the operating system, it is listed here to make it easier to verify the Performance Profile.

- In the MEASUREMENT STATISTICS column, examine the RUN MARGIN OF ERROR PCT and the CPU MARGIN OF ERROR PCT fields. A run-time margin of error of less than 2% reflects a reliable measurement. A high CPU margin of error does not mean the Performance Profile is invalid. Rather, it indicates that you should focus your analysis on the reports that detail run time.

Figure 3-1. Measurement Session Data Report

```
                                    ** MEASUREMENT SESSION DATA **
-------- JOB ENVIRONMENT --------    ------- MEASUREMENT PARAMETERS ------    ----- MEASUREMENT STATISTICS ----

PROGRAM MEASURED   -    BPXBATSL    ESTIMATED SESSION TIME  -       30 MIN    CPS TIME PERCENT -           43.32
JOB NAME           -      SXMUSS    TARGET SAMPLE SIZE      -       10,000    WAIT TIME PERCENT -          56.68
JOB NUMBER         -    JOB05182    REQUEST NUMBER   (Q)    -          156    RUN MARGIN OF ERROR PCT  -    1.00
STEP NAME   -                 GO                                             CPU MARGIN OF ERROR PCT  -    1.05

DATE OF SESSION    -  08/25/1999                                              TOTAL SAMPLES TAKEN -        8,611
TIME OF SESSION    -    17:21:04    SYS REQ -      PROD                       TOTAL SAMPLES PROCESSED -    8,611
CONDITION CODE     -      C-0000                                             INITIAL SAMPLING RATE-    5.56/SEC
                                    OPTIONS -         C                      FINAL SAMPLING RATE  -    5.56/SEC
SYSTEM -          OS/390 02.06.00
DFSMS              -       1.4.0                                             SESSION TIME -    30 MIN  1.54 SEC
SUBSYSTEM     -    UNIX SERVICES    --------- REPORT PARAMETERS ---------     CPU TIME -        11 MIN 11.97 SEC
CPU MODEL          -    9672-R42                                             WAIT TIME    -    14 MIN 38.02 SEC
SMF/SYSTEM ID -         PROD/PROD   REPORT RESOLUTION    -        2 BYTES    STRETCH TIME -     4 MIN 11.54 SEC
                                    SORTSIZE            -        999,999
REGION SIZE BELOW 16M  -  4,160K    LINES/PAGE          -             60    SRB TIME     -    0 MIN  28.67 SEC
REGION SIZE ABOVE  -     32,768K                                            SERVICE UNITS-             343,210
                                    DASD= 2.0%  DASDGAP= 5   ATTR= 0.0%
PTF LVL- 2.5.0.FS003021/FS003021                                            PAGES IN-       0     OUT-       0
STROBE TAPE NUMBER -  000-S23888    DATE FORMAT             -  MM/DD/YYYY   PAGING RATE  -            0.00/SEC
                                    TIME FORMAT             -    HH:MM:SS   EXCPS -        28,346     5.89/SEC
```

After you verify that the Performance Profile is valid, the next step is to determine which reports in the Performance Profile to analyze.

Choosing Between CPU and Wait Reports

You typically do not need to examine all of the reports in the Performance Profile. To determine whether to examine the CPU execution or wait reports in the Performance Profile, check the MEASUREMENT STATISTICS column and the values reported for CPS TIME PERCENT and the WAIT TIME PERCENT fields.

In Figure 3-1, the CPS TIME PERCENT value of 43.32% suggests that performance might be improved by reducing the CPU usage. CPU time is the run time during which the central processing was in use by application tasks executing within the measured job step. The WAIT TIME PERCENT value of 56.68% indicates that trying to curtail wait time might also improve performance.

In the traditional MVS environment, excessive I/O usually results in high wait; however, in the OS/390 UNIX System Services environment, excessive I/O can cause high CPU usage due to I/O processing taking place in OS/390 UNIX System Services address spaces. To determine whether the high CPU use is associated with data sets or CPU resources, first examine the Resource Demand Distribution report (Figure 3-2 on page 3-3). If the resource is CPU and the service by CPU is high, then review the execution reports described in this chapter to determine which programs are responsible for the high CPU use. If the high CPU use is associated with data sets, then review the Data Set Characteristics report and the Data Set Characteristics Supplement report as described in "Additional Reports To Examine" on page 3-8 and in Chapter 3 of *STROBE MVS Concepts and Facilities*.

In this chapter, the example focuses on reducing CPU usage associated with program activity. First we analyze the Resource Demand Distribution report to verify that program activity is responsible for the high CPU use. Next, we examine the following CPU execution reports:

- Program Section Usage Summary
- Program Section Usage By Procedure
- Attribution of CPU Execution Time

Identifying High CPU Use in an OS/390 UNIX System Services Environment

The first step in finding significant activity in an OS/390 UNIX System Services address space is to examine the Resource Demand Distribution report.

The Resource Demand Distribution report (Figure 3-2) lists the usage associated with specific tasks. In this example, more than 42% of the CPU use is associated with the program module **/u/s-ort**. The module **/u/s-ort** in the DDNAME or TASK column represents the initiating module. The percentage of 42% represents activity by the module itself and other modules that it called.

Note: In this case, the module name **/u/s-ort** is a *token*, which is an eight-byte STROBE-generated identifier. The token comprises the first four characters of the module or section name followed by a hyphen (-) and then the last three characters of the name. Refer to the Token - Longname Cross Reference report, Figure 3-6 on page 3-6, to determine the full name of the abbreviated token name.

Because more than 42% of the CPU use is associated with the program module **/u/s-ort**, this report indicates that program activity is responsible for high CPU use. The following example walks you through the CPU execution reports for a task causing high CPU usage. A detailed description of the Performance Profile is found in Chapter 3 of *STROBE MVS Concepts and Facilities*.

Figure 3-2. Resource Demand Distribution Report

** RESOURCE DEMAND DISTRIBUTION **

TASK OR DDNAME	RESOURCE	PERCENT OF RUN TIME SERVICED BY CPU	PERCENT OF RUN TIME SERVICED BY I/O	PERCENT OF RUN TIME SERVICED BY EITHER	PERCENT OF RUN TIME SPENT SOLO IN CPU	PERCENT OF RUN TIME SPENT SOLO IN I/O	PERCENT OF RUN TIME SPENT SOLO IN EITHER	PERCENT OF RUN TIME SPENT CAUSING CPU WAIT	CUMULATIVE PERCENTAGES SOLO TIME	CUMULATIVE PERCENTAGES CAUSING CPU WAIT
/u/s-ort	CPU	42.73	.00	42.73	42.68	.00	42.68	8.41	42.68	8.41
BPXBATSL	CPU	.51	.00	.51	.51	.00	.51	12.43	43.19	20.84
BPXPRATC	CPU	.03	.00	.03	.03	.00	.03	.08	43.22	20.92
.ZFS0003	3380	.03	26.89	26.92	.03	26.86	26.89	26.86	70.11	47.78
.HFS0002	3380	.03	8.92	8.95	.03	8.92	8.95	8.89	79.06	56.67
.FILEMGT		.05	.00	.05	.05	.00	.05	.00	79.11	56.67
.HFS0001	3380	.00	.03	.03	.00	.03	.03	.00	79.14	56.67

The Program Section Usage Summary Report

Once you have identified that program activity is responsible for high CPU usage, the next step is to identify the modules and control sections that are the greatest consumers of CPU by reviewing the Program Section Usage Summary report. To investigate a module's CPU consumption, first look at the TOTAL field under the % CPU TIME column and the histogram spike. In this example, the histogram spike indicates a CPU use of more than 86% (Figure 3-3 on page 3-4) for C370 system support routines within the pseudo-module .SYSTEM.

Note: The Program Section Usage Summary report groups all system routines by function under appropriately named pseudo-sections within a pseudo-module. Function-specific pseudo-sections, in turn, are treated as components of a pseudo-module.

The FUNCTION column shows that the routines are IBM C library subroutines. A high percentage of program activity taking place in the C370 library routines is common for a program running in OS/390 UNIX System Services. Next, refer to the Program Usage by Procedure report (Figure 3-4 on page 3-5) for the pseudo-module .SYSTEM to see which C370 library routines are consuming the most CPU resources.

Figure 3-3. Program Section Usage Summary Report

```
                                   ** PROGRAM SECTION USAGE SUMMARY **
 MODULE  SECTION 16M   SECT FUNCTION                          % CPU TIME   CPU TIME HISTOGRAM  MARGIN OF ERROR:  2.43%
   NAME  NAME    <,>   SIZE                                   SOLO   TOTAL   .00     22.00      44.00     66.00     88.00

.SYSTEM  .BPX              UNIX SYSTEM SERVICES              1.23    1.29   .
.SYSTEM  .CSR              CALLABLE SERVICES                  .06     .06   .
.SYSTEM  .C370LIB          IBM C LIBRARY SUBROUTINE         86.37   86.37   .****************************************
.SYSTEM  .GFU              HFS ADAPTER                        .43     .43   .
.SYSTEM  .IEF              ALLOCATE/UNALLOCATE                .12     .12   .
.SYSTEM  .IGW              DF/SMS                             .37     .37   .
.SYSTEM  .IRR              RACF                               .49     .49   .
.SYSTEM  .ISG              GRS                                .06     .06   .
.SYSTEM  .LELIB            LE/370 LIBRARY SUBROUTNE          4.97    5.03   .**
.SYSTEM  .PRIVATE          PRIVATE AREA                       .25     .25   .
.SYSTEM  .SVC              SUPERVISOR CONTROL                1.66    1.66   .
.SYSTEM  .VSAM             VIRTUAL STORAGE ACC METH           .06     .06   .
                                                            -----   -----
.SYSTEM  TOTALS            SYSTEM SERVICES                  96.07   96.19

/u/s-ort .LELIB        2481 LE/370 LIBRARY SUBROUTNE           .06     .06   .
/u/s-ort atof            10                                    .06     .06   .
/u/s-ort atoi            10                                    .12     .12   .
/u/s-ort fgets           10                                    .12     .12   .
/u/s-ort fprintf         10                                    .25     .25   .
/u/s-ort strtok          10                                    .18     .18   .
/u/s-ort CUSTOMER      1400                                   2.95    2.95   .*
                                                            -----   -----
/u/s-ort TOTALS   >    3931                                   3.74    3.74

BOPJCST           >    4408                                    .06     .06   .
                                                            -----   -----
 TOTALS                                                     99.88  100.00
```

The Program Usage by Procedure Report

The Program Usage by Procedure report (Figure 3-4 on page 3-5) details the time the CPU spent executing code within each control section of each module of the program or subsystem. In this example, the report shows CPU usage for all system service routines under the pseudo-module .SYSTEM. The header line identifies the pseudo-module (.SYSTEM) and the pseudo-section (.C370LIB). The detail lines show:

- MODULE NAME, the true module name or C library routines.
- SECTION NAME, the control section name (if STROBE obtained one during sampling).
- FUNCTION, the function descriptor of the control section (if available) or the function descriptor of the module.

In this Program Usage by Procedure report (Figure 3-4 on page 3-5), we see that more than 35% of the CPU time was associated with the IBM C/370 output routine EDCDFMTO. When system service routines identified in the Program Usage by Procedure report show high CPU time, examine the Attribution of CPU Execution Time report (Figure 3-5 on page 3-5) to pinpoint which application code lines invoked the system service routine.

Figure 3-4. Program Usage by Procedure Report

```
                                ** PROGRAM USAGE BY PROCEDURE **
        .SYSTEM     SYSTEM SERVICES                    .C370LIB  IBM C LIBRARY SUBROUTINE
 MODULE   SECTION        FUNCTION            INTERVAL          % CPU TIME    CPU TIME HISTOGRAM  MARGIN OF ERROR:  2.43%
   NAME      NAME                             LENGTH          SOLO   TOTAL   .00      9.00     18.00     27.00     36.00

 CEEEV003 @@GETR13 LE/370 C EVENT HANDLER          4           .25     .25   .
 CEEEV003 ATOI     CONV A STRING TO INTEGER      909          1.53    1.53   .*
 CEEEV003 CEETREN  TERMINATE ENCLAVE              20           .06     .06   .
 CEEEV003 DECHEX   DECIMAL TO HEXADECIMAL      2984          2.33    2.33   .**
 CEEEV003 EDCATOF  CONVERT STRING TO FLOAT      5944          7.61    7.61   .********
 CEEEV003 EDCATRM  TERM MEMBER REGISTRATION      548           .06     .06   .
 CEEEV003 EDCBCLSA CLOSE FILES AT PROG TERM     3972           .06     .06   .
 CEEEV003 EDCBGETS READ A STRING FROM FILE      1992          2.76    2.76   .***
 CEEEV003 EDCCTERM ENCLAVE TERMINATION          2948           .06     .06   .
 CEEEV003 EDCCTTRM THREAD TERMINATION           1164           .06     .06   .
 CEEEV003 EDCDFMTO FORMATTED OUTPUT ROUTINE    30756         35.48   35.48   .****************************************
 CEEEV003 EDCDOFCV FLOAT-PT CONVERSION RTNS     2696          7.86    7.86   .********
 CEEEV003 EDCFPRTF WRITE FORM O/P TO A FILE     5156          5.65    5.65   .******
 CEEEV003 EDCHCDTR DLL/C++ STATIC PROCESSNG     3592           .06     .06   .
 CEEEV003 EDCILG10 CALC INT OF LOG BASE 10       912          3.99    3.99   .****
 CEEEV003 EDCISBFP C/C++ RUNTIME MODULE         1192          5.34    5.34   .*****
 CEEEV003 EDCOTTRM C/C++ RUNTIME MODULE          868           .06     .06   .
 CEEEV003 EDCOXFR2 C/C++ RUNTIME MODULE        14728           .06     .06   .
 CEEEV003 EDCZMINV MAIN INVOC EVENT HANDLER      308           .06     .06   .
 CEEEV003 EDCZV003 C LIB LE EVENT HANDLER       2688           .06     .06   .
 CEEEV003 EDC1P124 I/O PROCESSING ROUTINES     12232           .18     .18   .
 CEEEV003 EDC1STOR IBM C LIBRARY SUBROUTINE     4412           .06     .06   .
 CEEEV003 HEXDEC   HEXADECIMAL TO DECIMAL       3516          7.37    7.37   .********
 CEEEV003 MVSFREES FREE STORAGE                  212           .06     .06   .
 CEEEV003 STRTOK   GET TOKEN FROM A STRING      2216          5.28    5.28   .*****
                                                             -----   -----
          .C370LIB  TOTALS                                  86.35   86.35
```

Identifying Performance Improvement Opportunities in the Source Code

The Attribution of CPU Execution Time report (Figure 3-5) identifies the lines of code or the offsets within the code that called the system routines and are therefore responsible for high CPU use. In this report, the IBM C/370 library routine CEEEV003 shows that it was invoked by the application module **/u/s-ort** and section name CUSTOMER.

Figure 3-5. Attribution of CPU Execution Time Report

```
                                ** ATTRIBUTION OF CPU EXECUTION TIME **

.C370LIB CEEEV003 EDCDFMTO  FORMATTED OUTPUT ROUTINE
 ---------------------------WAS INVOKED BY---------------------   ----------------VIA----------------------  CPU TIME %
 XACTION  MODULE   SECTION    RETURN   LINE   PROCEDURE NAME       MODULE   SECTION  FUNCTION                SOLO  TOTAL

          /u/s-ort CUSTOMER   000458     69      fprintf(output, " CEEEV003 EDCFPRTF WRITE FORM O/P TO A FILE 13.57 13.57
          /u/s-ort CUSTOMER   00048A     70      fprintf(output, " CEEEV003 EDCFPRTF WRITE FORM O/P TO A FILE 21.92 21.92
                                                                                                            ----- -----
                                                                                                           35.48 35.48
```

In this example, **/u/s-ort** is a token. STROBE generates a token when a module's name is longer than eight characters. A token comprises the first four characters of the module or section name followed by a hyphen (-) and then the last three characters of the name. Refer to the Token-Longname Cross Reference report (Figure 3-6 on page 3-6) to determine the longname of the token **/u/s-ort**.

Figure 3-6. Token - Longname Cross Reference Report

```
                          ** TOKEN - LONGNAME CROSS REFERENCE **
TOKEN     LONGNAME
/u/s-ort  /u/sxm/pubs/customer_inv_report
```

Now that we have determined the actual module name, our next step is to examine the Attribution of CPU Execution Time report (Figure 3-5 on page 3-5) to find the location where the application invoked the system service. Note that this Attribution of CPU Execution Time report is indexed; for the modules identified, it displays the source code line number and procedure name in addition to the hexadecimal starting location and interval length. For more information about indexed reports see Chapter 2 of *STROBE MVS Concepts and Facilities*.

In this example, the area to look for performance improvement opportunities in the source code is in module **/u/s-ort** at line 70.

Attributing CPU Execution Time Related to Source Code

An examination of the source code for line 70 (Figure 3-7) reveals that in control section CUSTOMER, the code is invoking the print routine that produces the investment calculation reports. While the print routine is using a significant amount of CPU, the routine is apparently performing a normal write function and cannot be changed to improve the application's performance. Instead, our next step is to examine the Program Usage By Procedure report (Figure 3-4 on page 3-5) again and locate another system service routine using a high amount of CPU.

Figure 3-7. Sample Code for Investment Calculation Print Routine

```
69                                                                                    69
70    27            fprintf(output, "%s %s %6f %6f\n", a, n, r, i);                   70
```

The Program Usage By Procedure report (Figure 3-4 on page 3-5) indicates that more than 7% of the CPU usage is associated with the IBM C/370 floating point conversion routine EDCDOFCV. Next, examine The Attribution of CPU Execution Time report (Figure 3-8 on page 3-7) to locate the area in the source code that invoked this subroutine.

The Attribution of CPU Execution Time report (Figure 3-8 on page 3-7) for the IBM C/370 floating point conversion routine EDCDOFCV shows that it was also invoked by the application module /u/s-ort and section name CUSTOMER at line 70. This routine is not only performing the normal write function as in our earlier example, but it is also performing floating point conversion. Since we know the write function cannot be changed, we should next examine the floating point conversion code for performance improvement opportunities.

Figure 3-8. Attribution of CPU Execution Time Report

```
                        ** ATTRIBUTION OF CPU EXECUTION TIME **

.C370LIB CEEEV003 EDCDOFCV  FLOAT-PT CONVERSION RTNS
 --------------------------WAS INVOKED BY---------------------   ------------------VIA-----------------------  CPU TIME %
 XACTION  MODULE   SECTION    RETURN   LINE   PROCEDURE NAME       MODULE   SECTION  FUNCTION                SOLO  TOTAL

          /u/s-ort CUSTOMER   000458     69      fprintf(output, " CEEEV003 EDCFPRTF WRITE FORM O/P TO A FILE  3.07  3.07
          /u/s-ort CUSTOMER   00048A     70      fprintf(output, " CEEEV003 EDCFPRTF WRITE FORM O/P TO A FILE  4.73  4.73
                                                                                                             ----- -----
                                                                                                              7.80  7.80
```

The floating point conversion routine embedded in the printf statement (Figure 3-7 on page 3-6) is performing a mathematical calculation for the final investment portfolio report. In this example, the variable definition of a, n, r, or i could be causing performance opportunities. Your next step is to see how the variables are defined.

In Figure 3-9, we see that variables r and i, as defined in lines 00025 and 00026 of the source listing, are handling double precision numeric values to complete the calculation task. Upon examination, it was determined that 32-digit arithmetic is not required for these calculations. The calculation was then changed to single precision integers, and a subsequent measurement of the application revealed a significant reduction in the overall CPU use associated with the application.

Figure 3-9. Variable Definition for IBM c/390 Floating Point Conversion Routine

```
00022        int  td;
00023        int  od;
00024        int  ed;
00025        double r;
00026        double i;
00027        char* today;
00028        time_t tt1;
00029        time_t tt2;
00030        double LengthPercent;
00031        double LengthFuture;
00032        double pv;
00033        double fv;
00034        double dif;
```

By comparing the Measurement Session Data reports for both measurements (Figure 3-1 and Figure 3-10) you can see a significant reduction in CPU session time, as well as an overall decrease in elapsed time.

By examining and comparing the CPU TIME fields, you see a total decrease in CPU time of almost two minutes.

Figure 3-10. Measurement Session Data Report (After)

```
                                    ** MEASUREMENT SESSION DATA **
-------- JOB ENVIRONMENT --------        ------- MEASUREMENT PARAMETERS ------        ----- MEASUREMENT STATISTICS ----

 PROGRAM MEASURED   -   BPXBATSL        ESTIMATED SESSION TIME  -       30 MIN       CPS TIME PERCENT -             40.36
 JOB NAME           -     SXMUSS        TARGET SAMPLE SIZE      -       10,000       WAIT TIME PERCENT -            59.64
 JOB NUMBER         -   JOB07657        REQUEST NUMBER (Q)      -          173       RUN MARGIN OF ERROR PCT  -      1.08
 STEP NAME   -              GO                                                      CPU MARGIN OF ERROR PCT  -      1.70

 DATE OF SESSION    - 08/29/1999                                                    TOTAL SAMPLES TAKEN -          8,180
 TIME OF SESSION    -   18:27:27        SYS REQ -      PROD                         TOTAL SAMPLES PROCESSED -      8,180
 CONDITION CODE     -     C-0000                                                   INITIAL SAMPLING RATE-      5.56/SEC
                                        OPTIONS -         C                       FINAL SAMPLING RATE  -      5.56/SEC
 SYSTEM -        OS/390 02.06.00
 DFSMS              -      1.4.0                                                   SESSION TIME -     28 MIN  32.41 SEC
 SUBSYSTEM      -   UNIX SERVICES        --------- REPORT PARAMETERS ---------     CPU TIME -          9 MIN  50.32 SEC
 CPU MODEL          -   9672-R42                                                   WAIT TIME    -     14 MIN  32.07 SEC
 SMF/SYSTEM ID -       PROD/PROD        REPORT RESOLUTION    -         2 BYTES     STRETCH TIME -      4 MIN  10.02 SEC
                                        SORTSIZE             -       999,999
 REGION SIZE BELOW 16M  - 4,160K        LINES/PAGE           -            60     SRB TIME     -      0 MIN  29.32 SEC
 REGION SIZE ABOVE  -    32,768K                                                 SERVICE UNITS-              343,210
                                        DASD= 2.0%     DASDGAP= 5  ATTR=  0.0%
 PTF LVL-2.5.0.FS003021/FS003021                                                 PAGES IN-             0     OUT-   0
 STROBE TAPE NUMBER - 000-S23888        DATE FORMAT            -   MM/DD/YYYY    PAGING RATE  -              0.00/SEC
                                        TIME FORMAT            -     HH:MM:SS    EXCPS -        28,346       5.89/SEC
```

Additional Reports To Examine

To streamline file access activities of OS/390 UNIX System Services routines, analyze the Data Set Characteristics and the Data Set Characteristics Supplement reports to identify the data sets exhibiting file access performance opportunities.

Data Set Characteristics Report

The Data Set Characteristics report shows the characteristics of all data sets accessed for the measured job step during the measurement session. The report provides the data set names associated with HFS or z/FS files listed on the Resource Demand Distribution report. For files that are allocated to the measured OS/390 task, the ddname will be the OS/390 ddname. For all other files, the ddname will be ".HFSnnnn" or ".ZFSnnnn, as shown in Figure 3-11.

Figure 3-11. 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

 .HFS0001 HFS                   4096                                        /u/sxm/pubs/customer_inv_report
 .ZFS0002 HFS                   4096                                    9608 /u/sxm/input.txt
 .HFS0003 HFS                   4096                                   18738 /u/sxm/output.txt
```

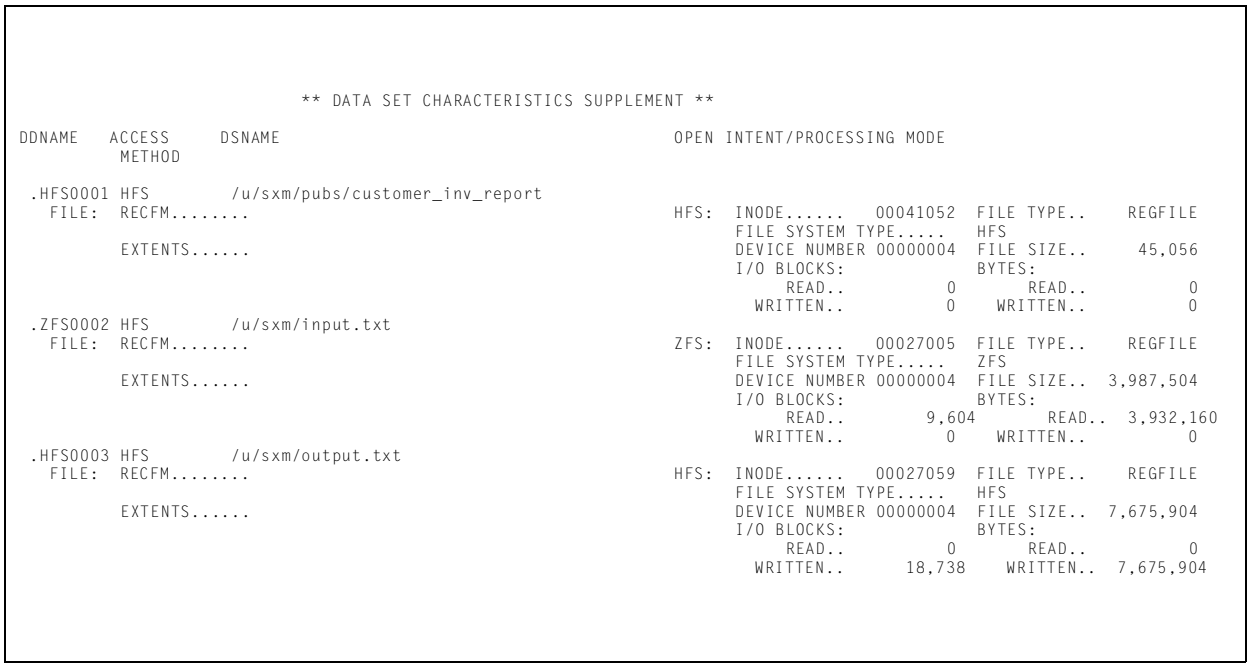
For HFS data sets, EXCP counts are collected only if the STROBE *STROBE UNIX System Services Feature* is installed at your site. Verify that the block size is optimal for access to the data stored upon the device.

Data Set Characteristics Supplement Report

The Data Set Characteristics Supplement Report provides additional information on data sets. You can cross-reference this report with the Data Set Characteristics report by data definition name, access method, and data set name. If no information exists, STROBE suppresses the names of the fields. If a data set is accessed from a HFS or z/FS system, the Data Set Characteristics Supplement Report details the following:

- INODE, displays the file identifier.
- FILE SYSTEM TYPE, the type of file system defined in the the system parmlib.
- DEVICE NUMBER, an indentifier for the mounted file system
- I/O BLOCKS, the number of blocks read from and written to the file during the measurement
- FILE TYPE, the definition of the file. This field can have one of two values: REGFILE for a normal file or FIFO for a named pipe
- BYTES, the number of bytes read from and written to the file during STROBE measurement

Figure 3-12. Data Set Characteristics Supplement Report



```
                          ** DATA SET CHARACTERISTICS SUPPLEMENT **

DDNAME   ACCESS     DSNAME                                        OPEN INTENT/PROCESSING MODE
          METHOD

 .HFS0001 HFS        /u/sxm/pubs/customer_inv_report
   FILE:  RECFM........                                    HFS:  INODE......   00041052  FILE TYPE..    REGFILE
                                                                 FILE SYSTEM TYPE.....   HFS
          EXTENTS......                                          DEVICE NUMBER 00000004  FILE SIZE..     45,056
                                                                 I/O BLOCKS:             BYTES:
                                                                      READ..          0       READ..          0
                                                                   WRITTEN..          0    WRITTEN..          0
 .ZFS0002 HFS        /u/sxm/input.txt
   FILE:  RECFM........                                    ZFS:  INODE......   00027005  FILE TYPE..    REGFILE
                                                                 FILE SYSTEM TYPE.....   ZFS
          EXTENTS......                                          DEVICE NUMBER 00000004  FILE SIZE..  3,987,504
                                                                 I/O BLOCKS:             BYTES:
                                                                      READ..      9,604       READ..  3,932,160
                                                                   WRITTEN..          0    WRITTEN..          0
 .HFS0003 HFS        /u/sxm/output.txt
   FILE:  RECFM........                                    HFS:  INODE......   00027059  FILE TYPE..    REGFILE
                                                                 FILE SYSTEM TYPE.....   HFS
          EXTENTS......                                          DEVICE NUMBER 00000004  FILE SIZE..  7,675,904
                                                                 I/O BLOCKS:             BYTES:
                                                                      READ..          0       READ..          0
                                                                   WRITTEN..     18,738    WRITTEN..  7,675,904
```

STROBE reports the I/O blocks and bytes statistics only if the System Management Facility (SMF) is configured to produce type 92 records.

Note: To avoid the recording of SMF type 92 record data, you can use an MVS system IEFU85 exit program. An example of the exit program is found in Appendix A of the *STROBE MVS System Programmer's Guide*.

To reduce file contention problems, refer to the I/O Facility Utilization Summary report to see what physical devices the files reside.

Index

N

O

P

Q

R

S

U