Harkins
Audit Software

REAL-TIME PROGRAM AUDIT™

RTPA for RPG

RTPA for CL

RTPA for COBOL

RTPA Query

United States Patent No. 6,775,827

Australian Patent – Patent No. 778165

User Manual

V4R3

V4R1M01

Harkins Audit Software, Inc.
816 Daisy Lane
West Chester, PA 19382
www.harkinsaudit.com

Telephone: 888.350.9148
610.431.1755
Fax: 610.436.1249

Technical Support: support@harkinsaudit.com
pharkins@harkinsaudit.com
paulhark@aol.com

Sales: sales@harkinsaudit.com
Paul Harkins email: paulhark@aol.com

RTPA Version: V4R3
Last Updated: 06/10/08
OS/400 Versions V5R3 through V6R1 (IBM Supported Releases)

Table of Contents

Preface

In writing this manual, we assume that you are familiar with the basics of programming RPGIII or RPGIV (RPGLE) on a System i (AS/400) computer. This manual is designed to help you get started with and use the Real-Time Program Audit™ for RPG software utility from Harkins Audit Software, Inc.

At a minimum, we hope that you will read:

Chapter 1: Installing Real-Time Program Audit to install the software
Chapter 4: Using RTPA to learn how to use RTPA
Appendix E: RTPA for RPG Examples of RTPA Audit Output

RTPA and Auditing: The Origins

Paul H. Harkins founded Harkins Audit Software to solve some of the most frustrating aspects of working as a software developer and IT consultant. Over his 45 year career, Paul came to realize that the most unpleasant, frustrating and time-consuming activities of a programmer involved:

- Learning other people's programs to make enhancements and corrections
- Tracking down bugs in production systems, and in development systems
- Validating and documenting new programs with proof of comprehensive testing
- The inability to easily and quickly solve difficult, critical, and stressful problems
- The Complex and unnecessary technical aspects of programming, particularly debugging
- Guessing and speculation of what happened or what might happen, if…
- Not being able to see exactly what was happening inside the computer all of the time
- The difficulty and loss of productivity in stopping the computer with traditional debug programs
- The lack of a recorded complete audit of what happened inside the computer programs
- The high risk and stress of supporting complex applications without any program audit trail

Over that time, the computer industry created a number of powerful tools and techniques to help with these activities. Programs like stepper-debuggers, 4th generation analysis tools and the structured walkthrough process improved the situation. At the same time, however, applications and many programs have become larger and more complicated, and more frequently involve teams of programmers; while at the same time, the productivity expectations for programmers have risen higher and higher.

In a moment of inspiration, when faced with a perfect example of these problems, Paul invented the Real-Time Program Auditing technique. Paul had been asked, as a consultant, to solve a bug in a piece of software that brought down a major warehouse, resulting in many thousands of dollars of lost time and contract and chargeback penalties. He was given a few minutes to find and fix the bug.

After determining that the bug could be anywhere in about 20,000 lines of legacy software (consisting of both in-house and 3rd party code), he decided that his best bet to find the bug was to cause the object program code to create a record of its execution (an "audit file") to allow him to see the program's actual execution flow. By comparing the times and data of the entries in the audit file with the time that the system would fail, and by reviewing the source statements that were actually executed including the data that was processed, he found the object program that was executing when the bug occurred and the program problem, allowing him to find and fix the bug much faster than anyone could have expected.

Paul realized that the only sure way of understanding exactly what was happening inside the computer was to be able to see and record, like a video security camera, every statement that was executed and all the data at each moment of time, without having to be present when the program was run.

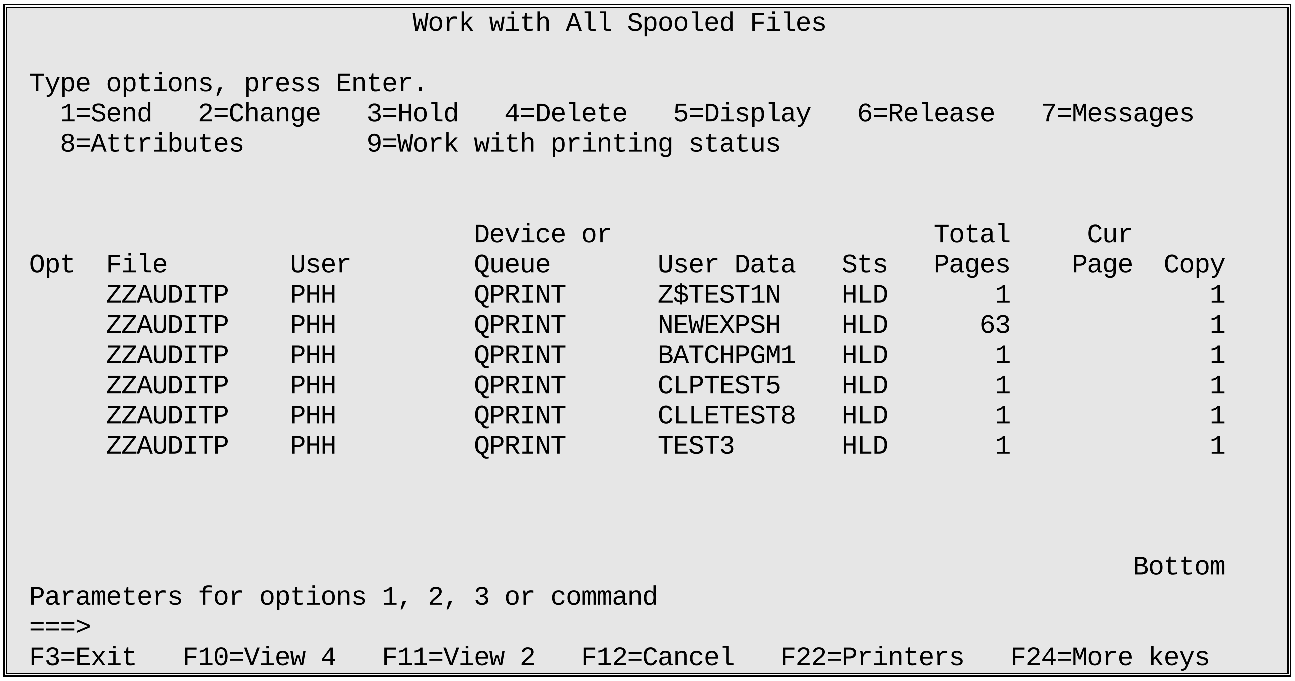
Paul took this technique and expanded it into the powerful tool that programmers all around the world use today. The Real-Time Program Audit gives programmers everywhere the ability to see, in complete detail, every line of source code that executes and the values of all the variables together with other information such as the time of execution, the change ID of the source, the source statement number, and the like. Programmers who have added RTPA to their collection of software tools have saved hours, days and months of development time, eliminating the tedium and frustration that comes from guessing about how a program actually executes, and greatly increasing their productivity.

RTPA Auditing of RPG, CL, and COBOL

The Real-Time Program Audit (RTPA) is designed to audit many programming languages, including the IBM System i computer languages; RPG (Report Program Generator), CL (Control Language), and COBOL (Common Business Oriented Language) languages. Both the Original Program Model (OPM) and the Integrated Language Environment (ILE) compilers for RPG, CLP and COBOL are supported.

The **RTPA** command is entered for expanding RPG source programs for auditing.
The **RTPACL** command is entered for expanding CL source programs for auditing.
The **RTPACO** command is entered for expanding COBOL source programs for auditing.
The **RTPAQ** command is entered for RTPA Query display and summarization of audit output

The RTPA audit output of executing source programs and the data for variables used in the audited executing source program statements id to the printer file ZZAUDITP. There is a separate ZZAUDITP spool (printer) file for each separate audited program. The WRKSPLF command is used to select and display the ZZAUDITP audit spool files for review.



```
                          Work with All Spooled Files

Type options, press Enter.
  1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
  8=Attributes        9=Work with printing status


                              Device or                         Total     Cur
Opt  File         User        Queue       User Data    Sts     Pages    Page   Copy
     ZZAUDITP     PHH         QPRINT      Z$TEST1N     HLD         1              1
     ZZAUDITP     PHH         QPRINT      NEWEXPSH     HLD        63              1
     ZZAUDITP     PHH         QPRINT      BATCHPGM1    HLD         1              1
     ZZAUDITP     PHH         QPRINT      CLPTEST5     HLD         1              1
     ZZAUDITP     PHH         QPRINT      CLLETEST8    HLD         1              1
     ZZAUDITP     PHH         QPRINT      TEST3        HLD         1              1



                                                                               Bottom
Parameters for options 1, 2, 3 or command
===>
F3=Exit   F10=View 4   F11=View 2   F12=Cancel   F22=Printers   F24=More keys
```

Figure P.1 RTPA Audit output ZZAUDITP created from the execution of call Z$TEST1N CLP

RTPA Query for PDF of all Job audit output by Execution Time

The Real-Time Program Audit RTPA Query product brings together all selected ZZAUDITP audit output spool files for a Job together in a spool file ZZAUDITS by the moment in time that the program was executed. This allows viewing the source statements and data executed in a logical job for all programs expanded for auditing, including CL, RPG, and COBOL programs, for all levels of program execution that job executes. Called programs expanded for auditing are shown no matter how many levels down in the call stack of the Job. The ZZAUDITS audit summary file, and the ZZAUDITP program audit spool files may be viewed in green screen with the WRKSPLF command or the RTPAQ (RTPA Query) command screen with Option 5. The ZZAUDITS audit summary file and the ZZAUDITP program audit files may be viewed in PDF format (Intranet only with limited number of pages unless **Adobe Acrobat Professional** is used) with the RTPAQ (RTPA Query) command with Option P.

RTPA Query ZZAUDITS audit summary output may also contain five cross References including:

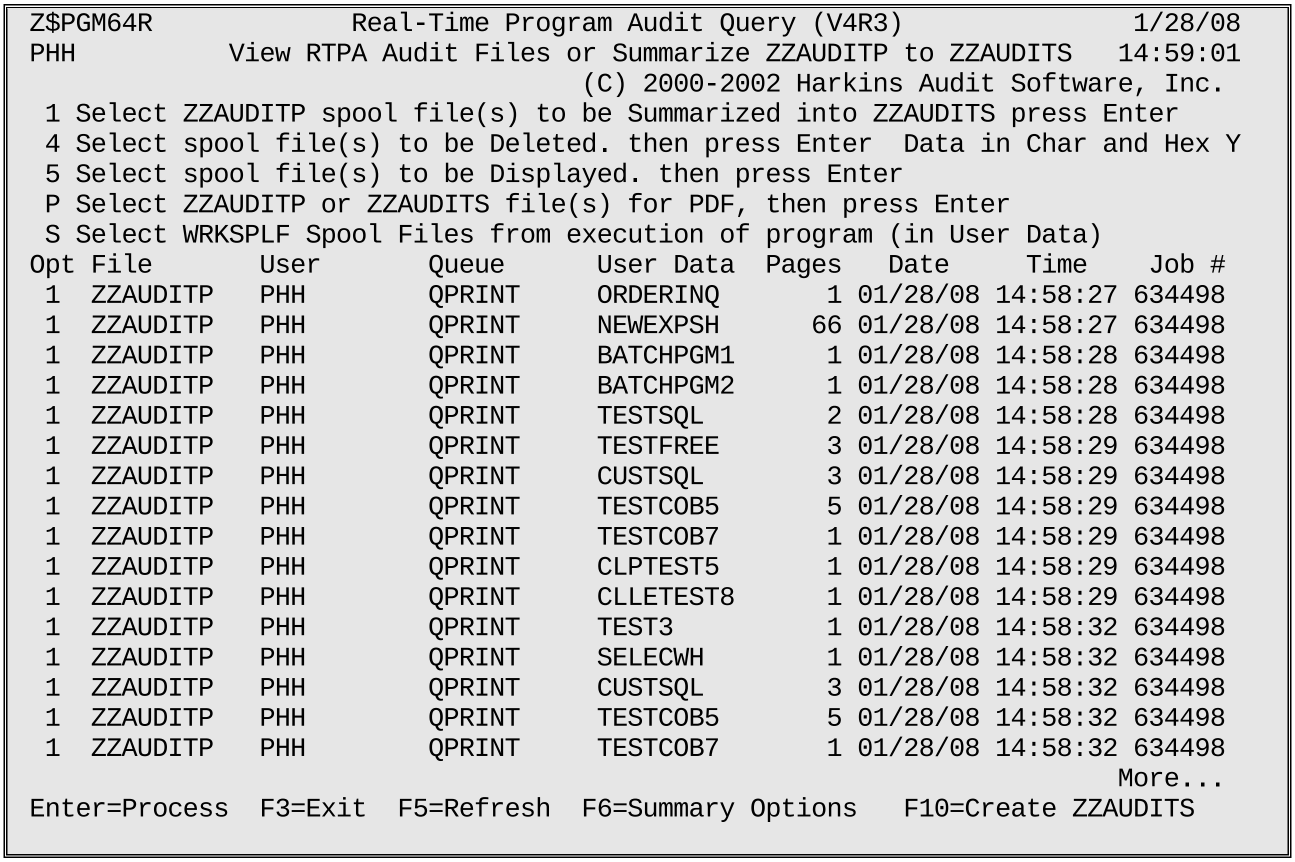
- **Fields- A cross reference of all fields, field attributes, file and library used in the audited program**
- **Files- A cross reference of all Files, file usage, record names, library used in the program**
- **Fileio- The program file I/O as actually used in the program (the first time a source statement is executed).**
- **Displays- The program WORKSTN (screen) file I/O statements and external definitions as actually used in the executing program (the first time a display source statement is executed). This illustrates the screens as created including the program variable names. Business Intelligence (BI) Metadata may be easily and**

correctly created using this information to create BI graphical GUI reports from key existing management displays.

- Reports- The program printer file I/O statements and printer file output variable and print positions as actually used in the executing program (the first time a printer source statement is executed). This illustrates the printer report(s) as created including the program variable names and the output print positions for both externally defined and internally defined (Output specifications) printer files. Business Intelligence (BI) Metadata may be easily and correctly created using this information to create BI graphical GUI reports from key existing management reports.
- Level- The program start time, program name, program type, and the program call stack level as actually executed in the executing logical job are displayed in the sequence the programs are initiated.

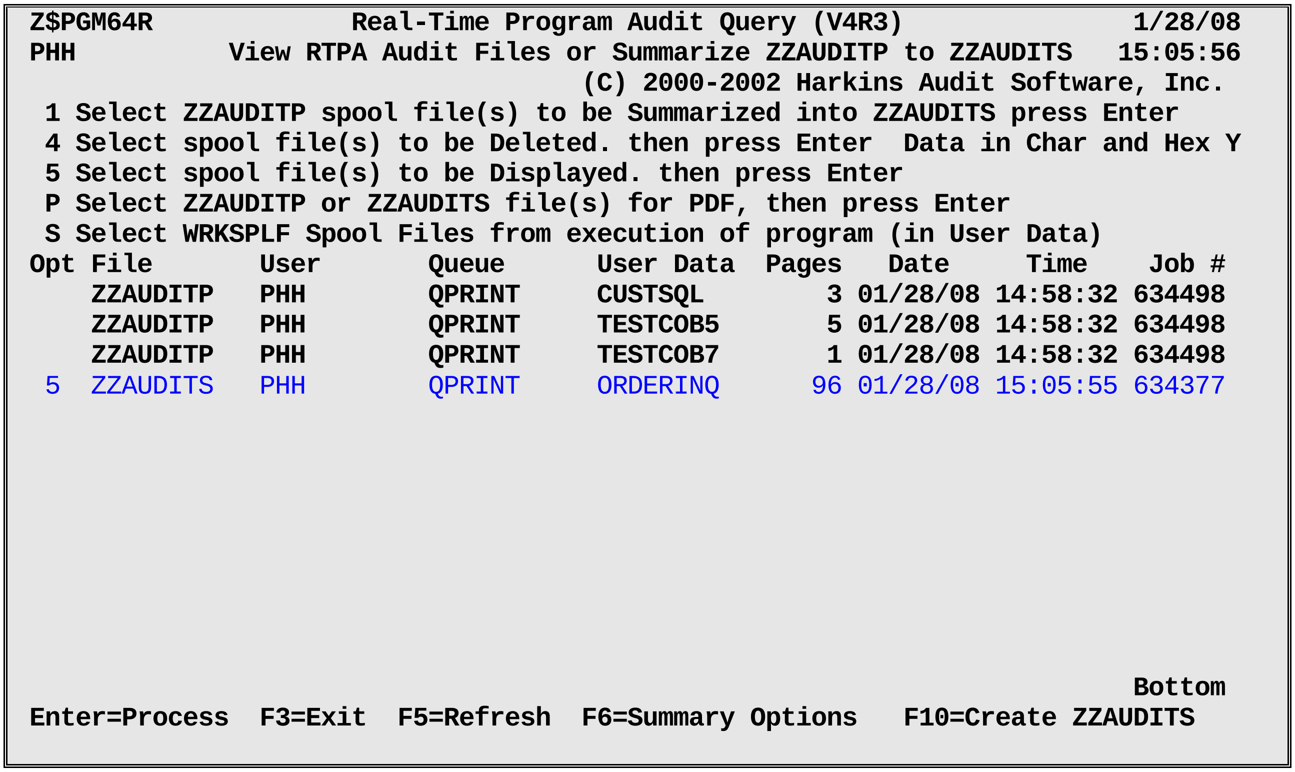
Additionally, the ZZAUDITS summary audit output and the ZZAUDITP audit output of a program execution may be searched by ANY full or partial data value or source statement executed.

- Program- Finds the first audit heading audit line for each program and the moment-of-time the program starts execution.
- File- Finds the next program file I/O statement that is executed in the program, including the File name, and the key field values and the record field values.
- Call- Finds the next program call statement to begin execution of the called program.
- Key- Finds the next program file I/O statement that is executed in the program, with Key fields and displays the key fields and data processed.
- Data- Finds the next program file I/O statement Data Area that is executed in the program.
- Printer- Finds the next program file Printer I/O statement that is executed in the program, including the printer file name, and the variable names and data printed by the program.
- Display- Finds the next program file Display Workstn I/O statement that is executed in the program, including the Workstn file and record name, and the variable names and data processed by the program.1
- SQL- Finds the next program embedded SQL statement set executed by the program.
- End- Finds the end of the program audit output.
- (character string) finds the next time the character string appears in the audit output is executed in the Green Screen WRKSPLF audit output
- (character string) finds all times the character string appears in the audit output is executed in the PDF audit output.



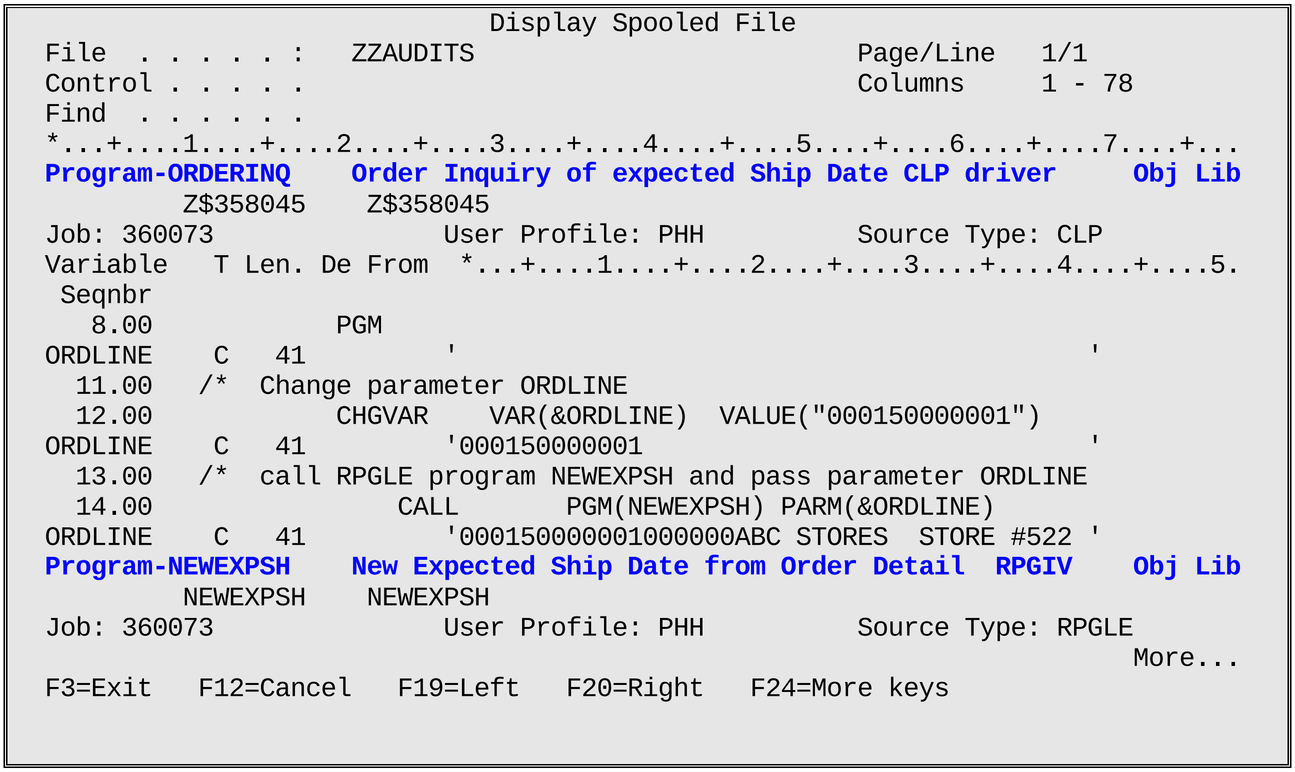
```
Z$PGM64R             Real-Time Program Audit Query (V4R3)              1/28/08
PHH          View RTPA Audit Files or Summarize ZZAUDITP to ZZAUDITS   14:59:01
                                    (C) 2000-2002 Harkins Audit Software, Inc.
 1 Select ZZAUDITP spool file(s) to be Summarized into ZZAUDITS press Enter
 4 Select spool file(s) to be Deleted. then press Enter  Data in Char and Hex Y
 5 Select spool file(s) to be Displayed. then press Enter
 P Select ZZAUDITP or ZZAUDITS file(s) for PDF, then press Enter
 S Select WRKSPLF Spool Files from execution of program (in User Data)
Opt File       User       Queue      User Data  Pages   Date     Time    Job #
 1  ZZAUDITP   PHH        QPRINT     ORDERINQ       1 01/28/08 14:58:27 634498
 1  ZZAUDITP   PHH        QPRINT     NEWEXPSH      66 01/28/08 14:58:27 634498
 1  ZZAUDITP   PHH        QPRINT     BATCHPGM1      1 01/28/08 14:58:28 634498
 1  ZZAUDITP   PHH        QPRINT     BATCHPGM2      1 01/28/08 14:58:28 634498
 1  ZZAUDITP   PHH        QPRINT     TESTSQL        2 01/28/08 14:58:28 634498
 1  ZZAUDITP   PHH        QPRINT     TESTFREE       3 01/28/08 14:58:29 634498
 1  ZZAUDITP   PHH        QPRINT     CUSTSQL        3 01/28/08 14:58:29 634498
 1  ZZAUDITP   PHH        QPRINT     TESTCOB5       5 01/28/08 14:58:29 634498
 1  ZZAUDITP   PHH        QPRINT     TESTCOB7       1 01/28/08 14:58:29 634498
 1  ZZAUDITP   PHH        QPRINT     CLPTEST5       1 01/28/08 14:58:29 634498
 1  ZZAUDITP   PHH        QPRINT     CLLETEST8      1 01/28/08 14:58:29 634498
 1  ZZAUDITP   PHH        QPRINT     TEST3          1 01/28/08 14:58:32 634498
 1  ZZAUDITP   PHH        QPRINT     SELECWH        1 01/28/08 14:58:32 634498
 1  ZZAUDITP   PHH        QPRINT     CUSTSQL        3 01/28/08 14:58:32 634498
 1  ZZAUDITP   PHH        QPRINT     TESTCOB5       5 01/28/08 14:58:32 634498
 1  ZZAUDITP   PHH        QPRINT     TESTCOB7       1 01/28/08 14:58:32 634498
                                                                       More...
Enter=Process  F3=Exit  F5=Refresh  F6=Summary Options   F10=Create ZZAUDITS
```

Figure P.2 RTPA Query display of ZZAUDITP Audited Programs F10 for RTPA Query Summary



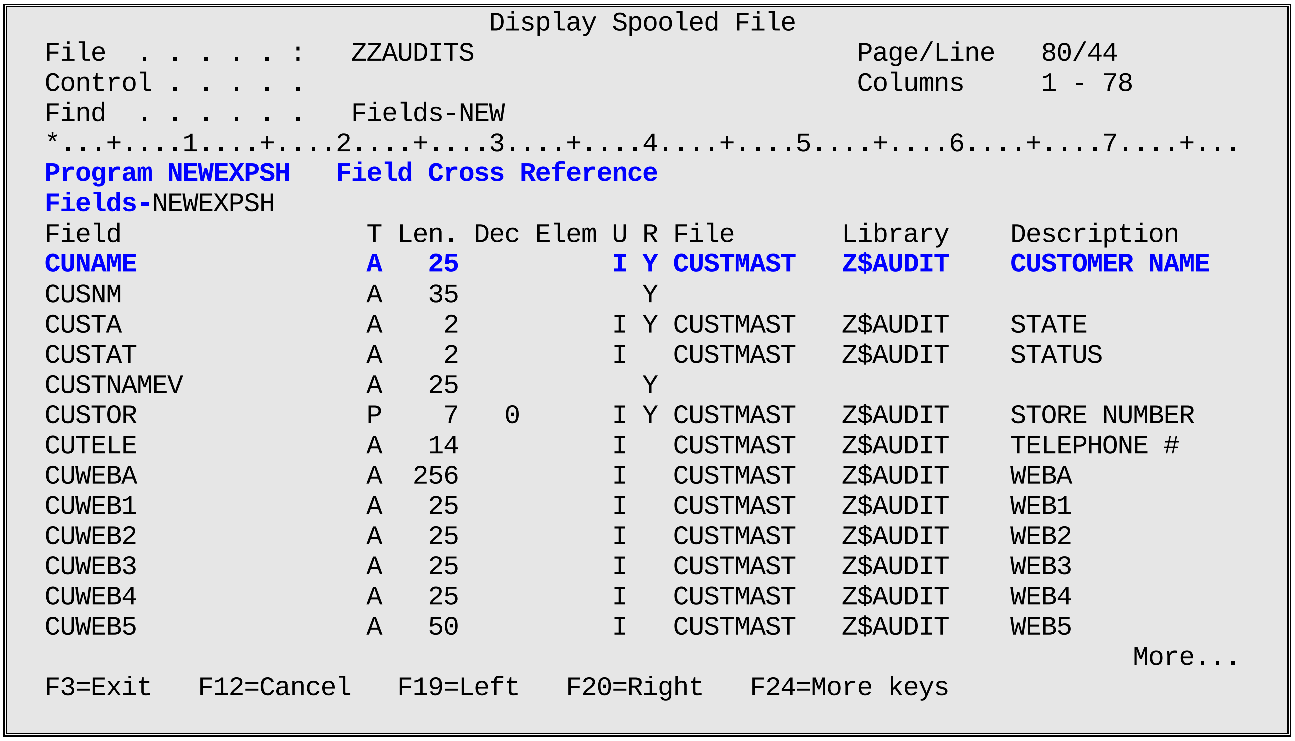
```
Z$PGM64R             Real-Time Program Audit Query (V4R3)              1/28/08
PHH          View RTPA Audit Files or Summarize ZZAUDITP to ZZAUDITS   15:05:56
                                    (C) 2000-2002 Harkins Audit Software, Inc.
 1 Select ZZAUDITP spool file(s) to be Summarized into ZZAUDITS press Enter
 4 Select spool file(s) to be Deleted. then press Enter  Data in Char and Hex Y
 5 Select spool file(s) to be Displayed. then press Enter
 P Select ZZAUDITP or ZZAUDITS file(s) for PDF, then press Enter
 S Select WRKSPLF Spool Files from execution of program (in User Data)
Opt File       User       Queue      User Data  Pages   Date     Time    Job #
    ZZAUDITP   PHH        QPRINT     CUSTSQL        3 01/28/08 14:58:32 634498
    ZZAUDITP   PHH        QPRINT     TESTCOB5       5 01/28/08 14:58:32 634498
    ZZAUDITP   PHH        QPRINT     TESTCOB7       1 01/28/08 14:58:32 634498
 5  ZZAUDITS   PHH        QPRINT     ORDERINQ      96 01/28/08 15:05:55 634377















                                                                        Bottom
Enter=Process  F3=Exit  F5=Refresh  F6=Summary Options   F10=Create ZZAUDITS
```

Figure P3 RTPA Query display of ZZAUDITS Audited Summary file ZZAUDITS



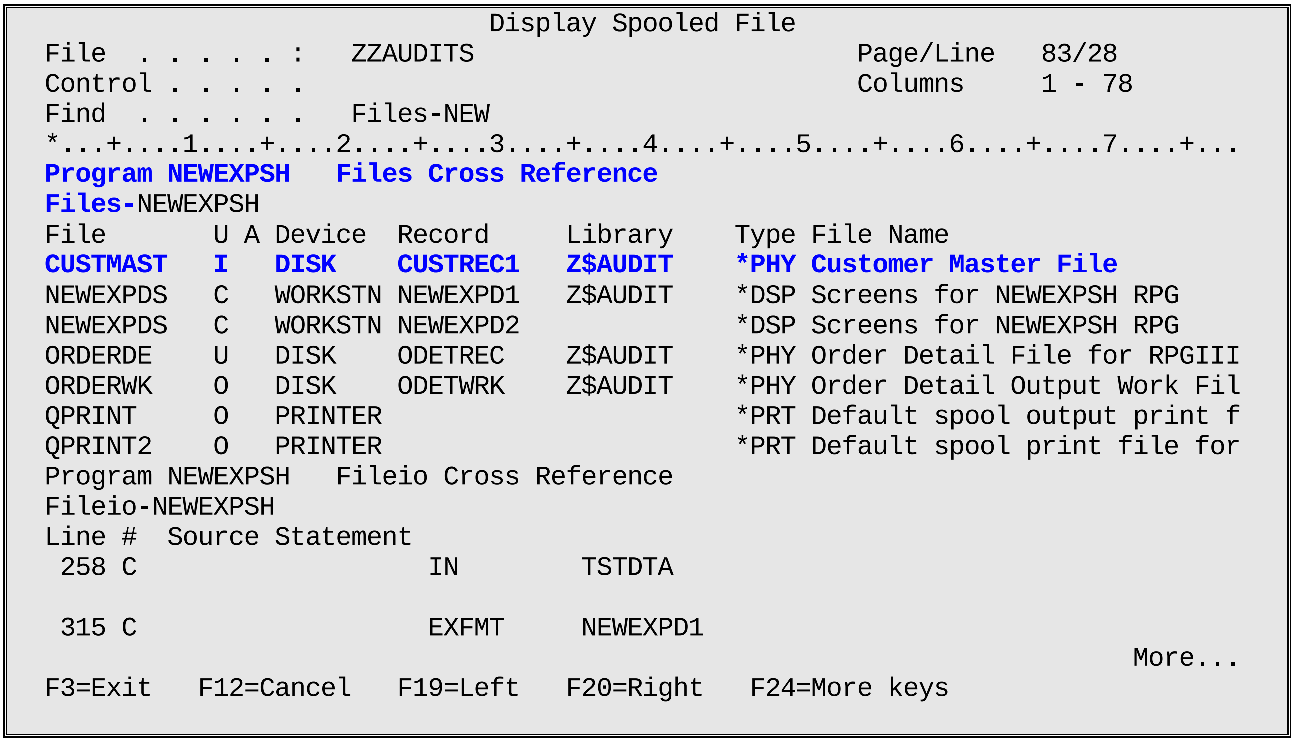
```
                             Display Spooled File
File  . . . . . :   ZZAUDITS                         Page/Line   1/1
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program-ORDERINQ    Order Inquiry of expected Ship Date CLP driver     Obj Lib
         Z$358045    Z$358045
Job: 360073               User Profile: PHH          Source Type: CLP
Variable   T Len. De From  *...+....1....+....2....+....3....+....4....+....5.
 Seqnbr
   8.00            PGM
ORDLINE    C   41          '                                          '
  11.00   /*  Change parameter ORDLINE
  12.00            CHGVAR    VAR(&ORDLINE)  VALUE("000150000001")
ORDLINE    C   41          '000150000001                              '
  13.00   /*  call RPGLE program NEWEXPSH and pass parameter ORDLINE
  14.00                CALL       PGM(NEWEXPSH) PARM(&ORDLINE)
ORDLINE    C   41          '0001500000001000000ABC STORES  STORE #522 '
Program-NEWEXPSH    New Expected Ship Date from Order Detail  RPGIV    Obj Lib
         NEWEXPSH    NEWEXPSH
Job: 360073               User Profile: PHH          Source Type: RPGLE
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure P.4 RTPA Query display of audit output for an entire logical job by execution time



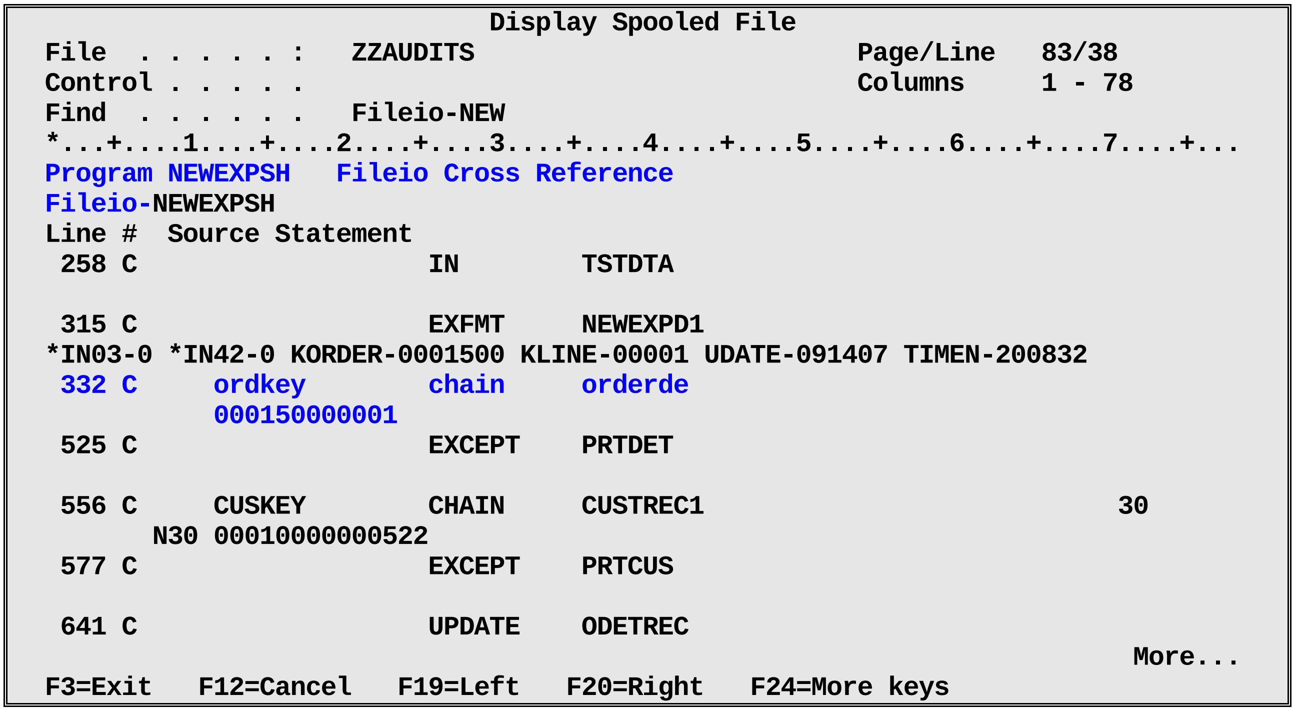
```
                             Display Spooled File
File  . . . . . :   ZZAUDITS                         Page/Line   80/44
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .   Fields-NEW
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program NEWEXPSH   Field Cross Reference
Fields-NEWEXPSH
Field                T Len. Dec Elem U R File       Library    Description
CUNAME               A    25         I Y CUSTMAST   Z$AUDIT    CUSTOMER NAME
CUSNM                A    35           Y
CUSTA                A     2         I Y CUSTMAST   Z$AUDIT    STATE
CUSTAT               A     2         I   CUSTMAST   Z$AUDIT    STATUS
CUSTNAMEV            A    25           Y
CUSTOR               P     7   0     I Y CUSTMAST   Z$AUDIT    STORE NUMBER
CUTELE               A    14         I   CUSTMAST   Z$AUDIT    TELEPHONE #
CUWEBA               A   256         I   CUSTMAST   Z$AUDIT    WEBA
CUWEB1               A    25         I   CUSTMAST   Z$AUDIT    WEB1
CUWEB2               A    25         I   CUSTMAST   Z$AUDIT    WEB2
CUWEB3               A    25         I   CUSTMAST   Z$AUDIT    WEB3
CUWEB4               A    25         I   CUSTMAST   Z$AUDIT    WEB4
CUWEB5               A    50         I   CUSTMAST   Z$AUDIT    WEB5
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure P.5 RTPA Query display of an Audited Program Field Cross Reference



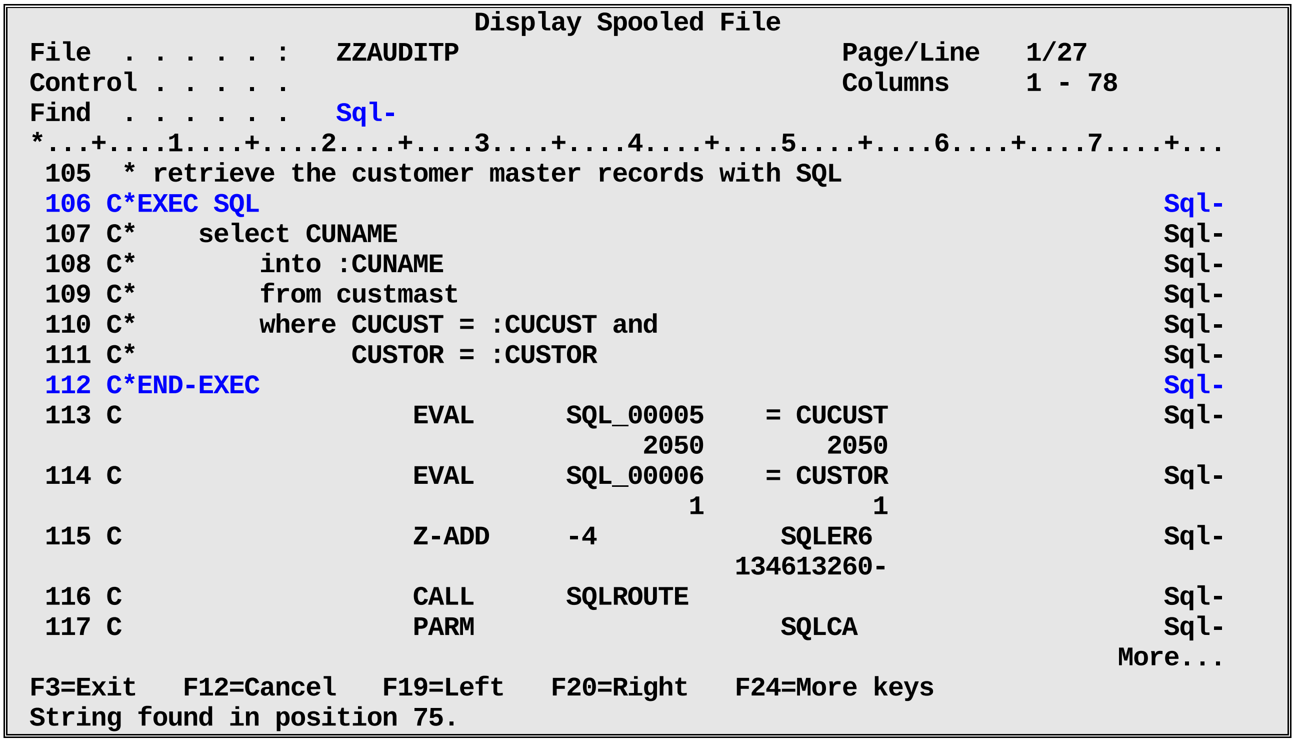
```
                             Display Spooled File
File  . . . . . :   ZZAUDITS                         Page/Line   83/28
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .   Files-NEW
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program NEWEXPSH   Files Cross Reference
Files-NEWEXPSH
File       U A Device  Record     Library    Type File Name
CUSTMAST   I   DISK    CUSTREC1   Z$AUDIT    *PHY Customer Master File
NEWEXPDS   C   WORKSTN NEWEXPD1   Z$AUDIT    *DSP Screens for NEWEXPSH RPG
NEWEXPDS   C   WORKSTN NEWEXPD2              *DSP Screens for NEWEXPSH RPG
ORDERDE    U   DISK    ODETREC    Z$AUDIT    *PHY Order Detail File for RPGIII
ORDERWK    O   DISK    ODETWRK    Z$AUDIT    *PHY Order Detail Output Work Fil
QPRINT     O   PRINTER                       *PRT Default spool output print f
QPRINT2    O   PRINTER                       *PRT Default spool print file for
Program NEWEXPSH   Fileio Cross Reference
Fileio-NEWEXPSH
Line #  Source Statement
 258 C                  IN         TSTDTA

 315 C                  EXFMT      NEWEXPD1
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure P.6 RTPA Query display of an Audited Program Files Cross Reference



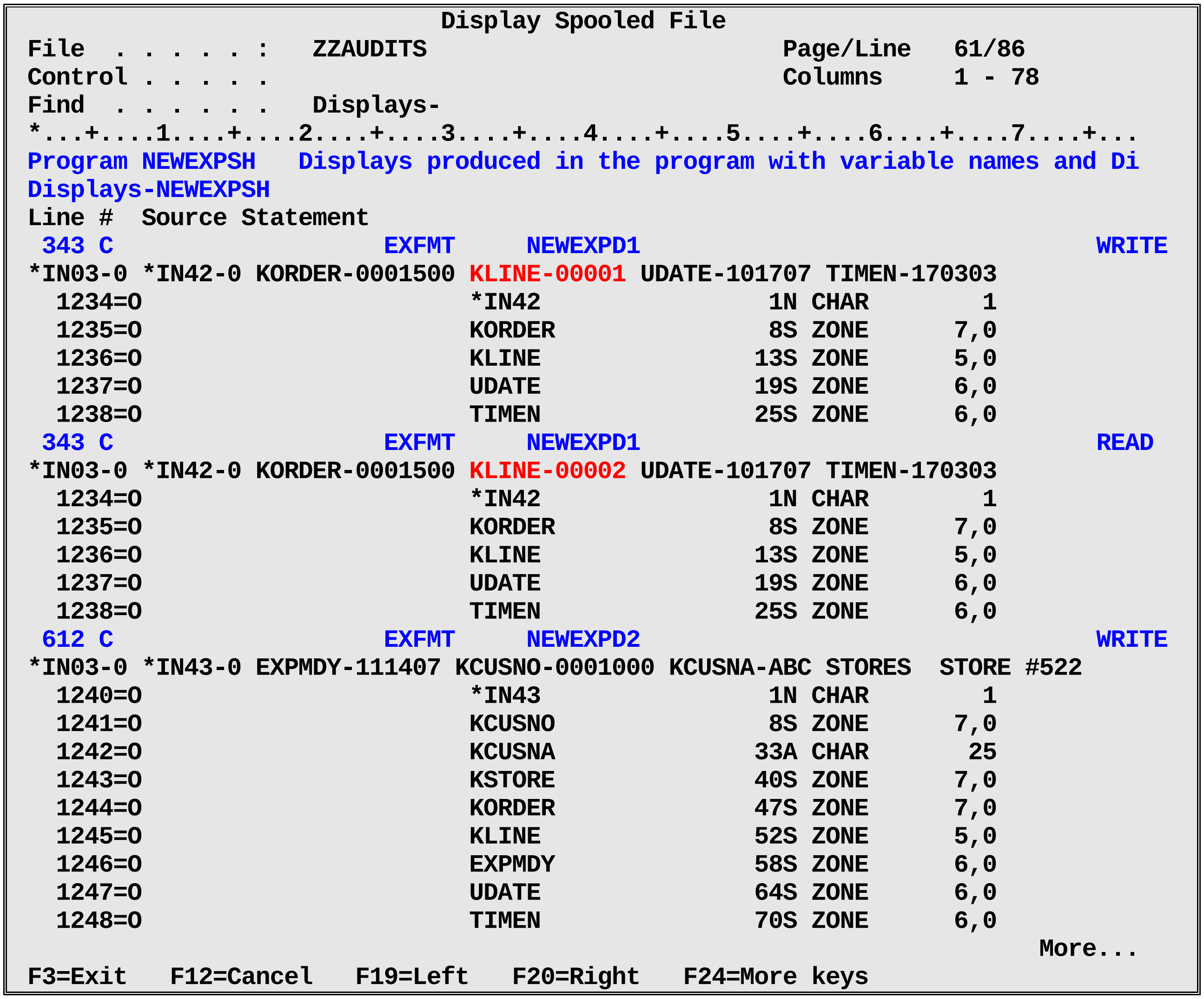
```
                             Display Spooled File
File  . . . . . :   ZZAUDITS                         Page/Line   83/38
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .   Fileio-NEW
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program NEWEXPSH   Fileio Cross Reference
Fileio-NEWEXPSH
Line #  Source Statement
 258 C                  IN         TSTDTA

 315 C                  EXFMT      NEWEXPD1
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-091407 TIMEN-200832
 332 C     ordkey        chain      orderde
           000150000001
 525 C                  EXCEPT     PRTDET

 556 C     CUSKEY        CHAIN      CUSTREC1                           30
       N30 00010000000522
 577 C                  EXCEPT     PRTCUS

 641 C                  UPDATE     ODETREC
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure P.7 RTPA Query display of an Audited Program Fileio Cross Reference



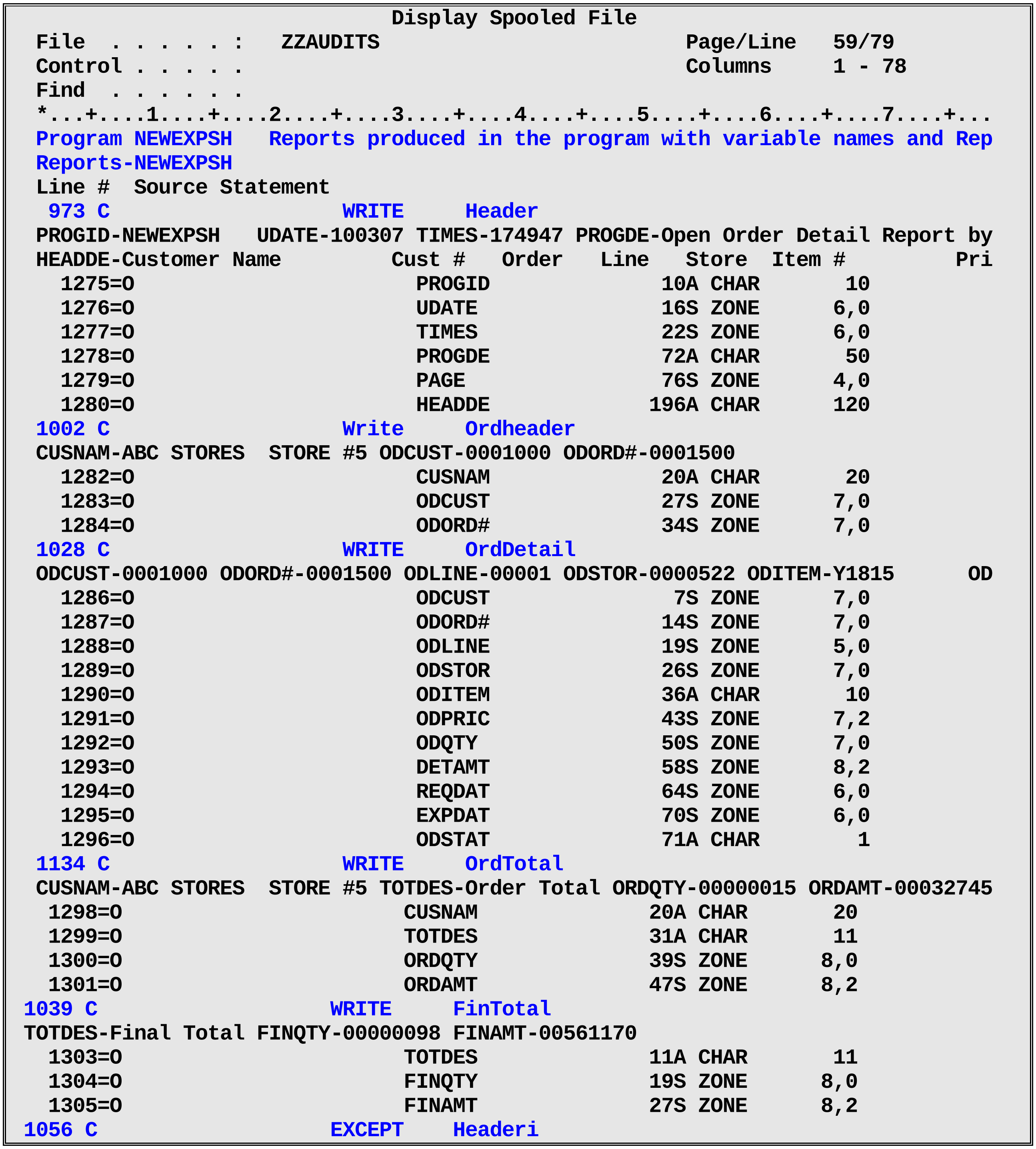
```
                             Display Spooled File
File  . . . . . :   ZZAUDITP                         Page/Line   1/27
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .   Sql-
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 105   * retrieve the customer master records with SQL
 106 C*EXEC SQL                                                           Sql-
 107 C*    select CUNAME                                                  Sql-
 108 C*        into :CUNAME                                               Sql-
 109 C*        from custmast                                              Sql-
 110 C*        where CUCUST = :CUCUST and                                 Sql-
 111 C*              CUSTOR = :CUSTOR                                     Sql-
 112 C*END-EXEC                                                           Sql-
 113 C                  EVAL      SQL_00005    = CUCUST                   Sql-
                                       2050         2050
 114 C                  EVAL      SQL_00006    = CUSTOR                   Sql-
                                          1            1
 115 C                  Z-ADD     -4            SQLER6                    Sql-
                                             134613260-
 116 C                  CALL      SQLROUTE                                Sql-
 117 C                  PARM                    SQLCA                     Sql-
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
String found in position 75.
```

Figure P.8 RTPA Query display of SQL source statements, and the generated SQL code



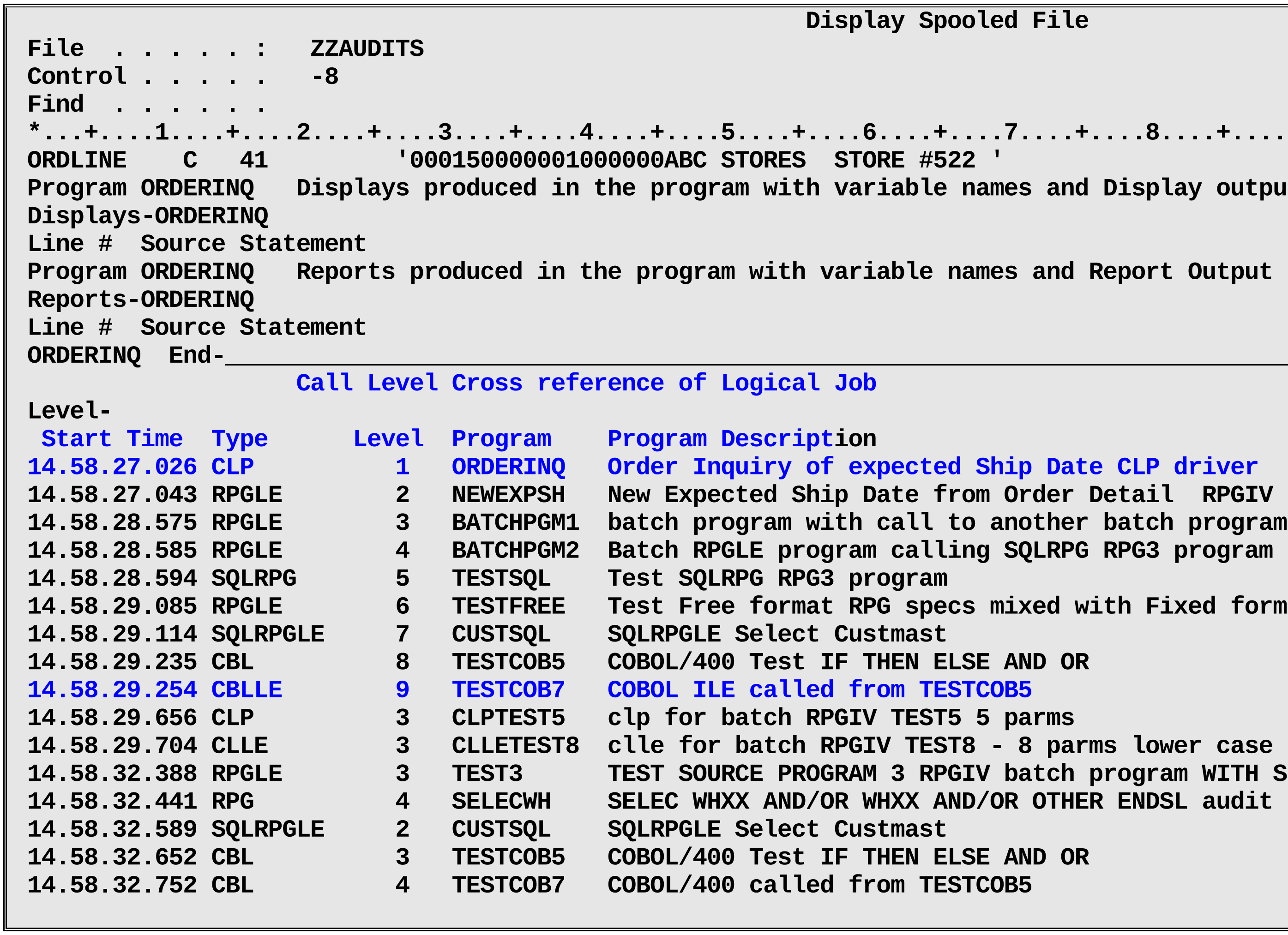
```
                          Display Spooled File
File  . . . . . :   ZZAUDITS                         Page/Line   61/86
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .   Displays-
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program NEWEXPSH   Displays produced in the program with variable names and Di
Displays-NEWEXPSH
Line #  Source Statement
 343 C                  EXFMT     NEWEXPD1                            WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-101707 TIMEN-170303
  1234=O                       *IN42               1N CHAR        1
  1235=O                       KORDER              8S ZONE      7,0
  1236=O                       KLINE              13S ZONE      5,0
  1237=O                       UDATE              19S ZONE      6,0
  1238=O                       TIMEN              25S ZONE      6,0
 343 C                  EXFMT     NEWEXPD1                            READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-101707 TIMEN-170303
  1234=O                       *IN42               1N CHAR        1
  1235=O                       KORDER              8S ZONE      7,0
  1236=O                       KLINE              13S ZONE      5,0
  1237=O                       UDATE              19S ZONE      6,0
  1238=O                       TIMEN              25S ZONE      6,0
 612 C                  EXFMT     NEWEXPD2                            WRITE
*IN03-0 *IN43-0 EXPMDY-111407 KCUSNO-0001000 KCUSNA-ABC STORES  STORE #522
  1240=O                       *IN43               1N CHAR        1
  1241=O                       KCUSNO              8S ZONE      7,0
  1242=O                       KCUSNA             33A CHAR       25
  1243=O                       KSTORE             40S ZONE      7,0
  1244=O                       KORDER             47S ZONE      7,0
  1245=O                       KLINE              52S ZONE      5,0
  1246=O                       EXPMDY             58S ZONE      6,0
  1247=O                       UDATE              64S ZONE      6,0
  1248=O                       TIMEN              70S ZONE      6,0
                                                                    More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure P.9 RTPA Query display of Displays (screens) by the program with variable names and data



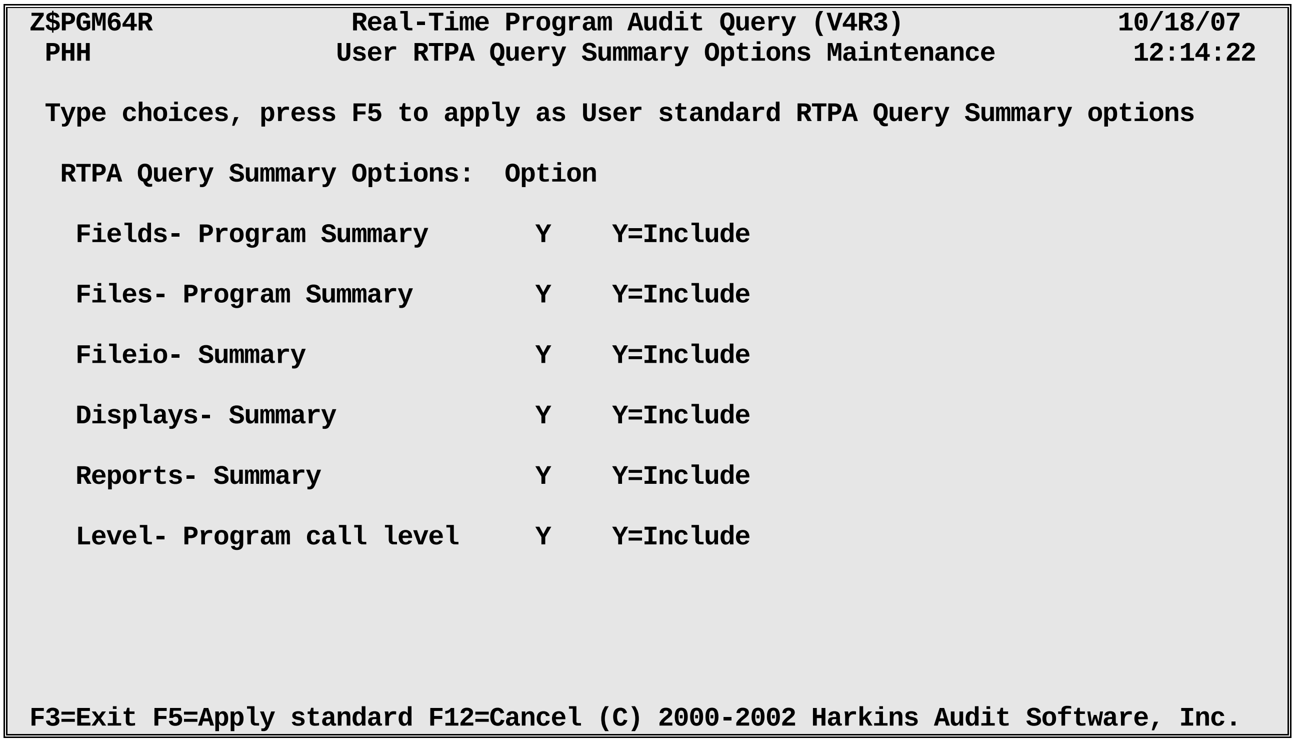
```
                              Display Spooled File
 File  . . . . . :   ZZAUDITS                        Page/Line   59/79
 Control . . . . .                                   Columns     1 - 78
 Find  . . . . . .
 *...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 Program NEWEXPSH   Reports produced in the program with variable names and Rep
 Reports-NEWEXPSH
 Line #  Source Statement
  973 C                  WRITE     Header
 PROGID-NEWEXPSH   UDATE-100307 TIMES-174947 PROGDE-Open Order Detail Report by
 HEADDE-Customer Name         Cust #   Order   Line   Store  Item #         Pri
   1275=O                        PROGID              10A CHAR       10
   1276=O                        UDATE               16S ZONE      6,0
   1277=O                        TIMES               22S ZONE      6,0
   1278=O                        PROGDE              72A CHAR       50
   1279=O                        PAGE                76S ZONE      4,0
   1280=O                        HEADDE             196A CHAR      120
 1002 C                  Write     Ordheader
 CUSNAM-ABC STORES  STORE #5 ODCUST-0001000 ODORD#-0001500
   1282=O                        CUSNAM              20A CHAR       20
   1283=O                        ODCUST              27S ZONE      7,0
   1284=O                        ODORD#              34S ZONE      7,0
 1028 C                  WRITE     OrdDetail
 ODCUST-0001000 ODORD#-0001500 ODLINE-00001 ODSTOR-0000522 ODITEM-Y1815      OD
   1286=O                        ODCUST               7S ZONE      7,0
   1287=O                        ODORD#              14S ZONE      7,0
   1288=O                        ODLINE              19S ZONE      5,0
   1289=O                        ODSTOR              26S ZONE      7,0
   1290=O                        ODITEM              36A CHAR       10
   1291=O                        ODPRIC              43S ZONE      7,2
   1292=O                        ODQTY               50S ZONE      7,0
   1293=O                        DETAMT              58S ZONE      8,2
   1294=O                        REQDAT              64S ZONE      6,0
   1295=O                        EXPDAT              70S ZONE      6,0
   1296=O                        ODSTAT              71A CHAR        1
 1134 C                  WRITE     OrdTotal
 CUSNAM-ABC STORES  STORE #5 TOTDES-Order Total ORDQTY-00000015 ORDAMT-00032745
  1298=O                        CUSNAM              20A CHAR       20
  1299=O                        TOTDES              31A CHAR       11
  1300=O                        ORDQTY              39S ZONE      8,0
  1301=O                        ORDAMT              47S ZONE      8,2
1039 C                   WRITE     FinTotal
TOTDES-Final Total FINQTY-00000098 FINAMT-00561170
  1303=O                        TOTDES              11A CHAR       11
  1304=O                        FINQTY              19S ZONE      8,0
  1305=O                        FINAMT              27S ZONE      8,2
1056 C                   EXCEPT    Headeri
```

Figure P.10 RTPA Query display of Reports produced by the program with variable names and data



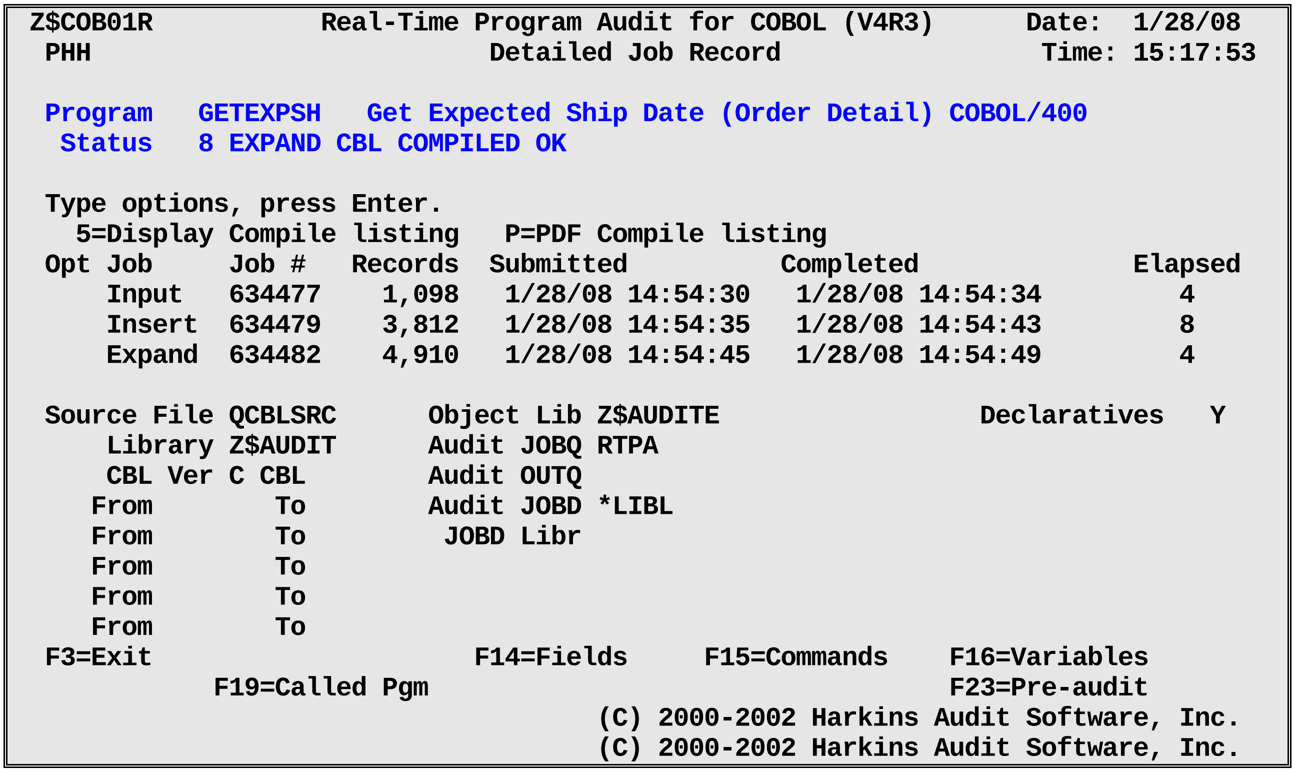
```
                                                    Display Spooled File
File  . . . . . :   ZZAUDITS
Control . . . . .   -8
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....
ORDLINE    C   41         '0001500000010000000ABC STORES  STORE #522 '
Program ORDERINQ   Displays produced in the program with variable names and Display outpu
Displays-ORDERINQ
Line #  Source Statement
Program ORDERINQ   Reports produced in the program with variable names and Report Output
Reports-ORDERINQ
Line #  Source Statement
ORDERINQ  End-____________________________________________________________________________
                   Call Level Cross reference of Logical Job
Level-
 Start Time  Type      Level  Program    Program Description
14.58.27.026 CLP           1  ORDERINQ   Order Inquiry of expected Ship Date CLP driver
14.58.27.043 RPGLE         2  NEWEXPSH   New Expected Ship Date from Order Detail  RPGIV
14.58.28.575 RPGLE         3  BATCHPGM1  batch program with call to another batch program
14.58.28.585 RPGLE         4  BATCHPGM2  Batch RPGLE program calling SQLRPG RPG3 program
14.58.28.594 SQLRPG        5  TESTSQL    Test SQLRPG RPG3 program
14.58.29.085 RPGLE         6  TESTFREE   Test Free format RPG specs mixed with Fixed form
14.58.29.114 SQLRPGLE      7  CUSTSQL    SQLRPGLE Select Custmast
14.58.29.235 CBL           8  TESTCOB5   COBOL/400 Test IF THEN ELSE AND OR
14.58.29.254 CBLLE         9  TESTCOB7   COBOL ILE called from TESTCOB5
14.58.29.656 CLP           3  CLPTEST5   clp for batch RPGIV TEST5 5 parms
14.58.29.704 CLLE          3  CLLETEST8  clle for batch RPGIV TEST8 - 8 parms lower case
14.58.32.388 RPGLE         3  TEST3      TEST SOURCE PROGRAM 3 RPGIV batch program WITH S
14.58.32.441 RPG           4  SELECWH    SELEC WHXX AND/OR WHXX AND/OR OTHER ENDSL audit
14.58.32.589 SQLRPGLE      2  CUSTSQL    SQLRPGLE Select Custmast
14.58.32.652 CBL           3  TESTCOB5   COBOL/400 Test IF THEN ELSE AND OR
14.58.32.752 CBL           4  TESTCOB7   COBOL/400 called from TESTCOB5
```

Figure P.11 RTPA Query display of ZZAUDITS Audit Summary Program Call Stack Level – 9 Levels



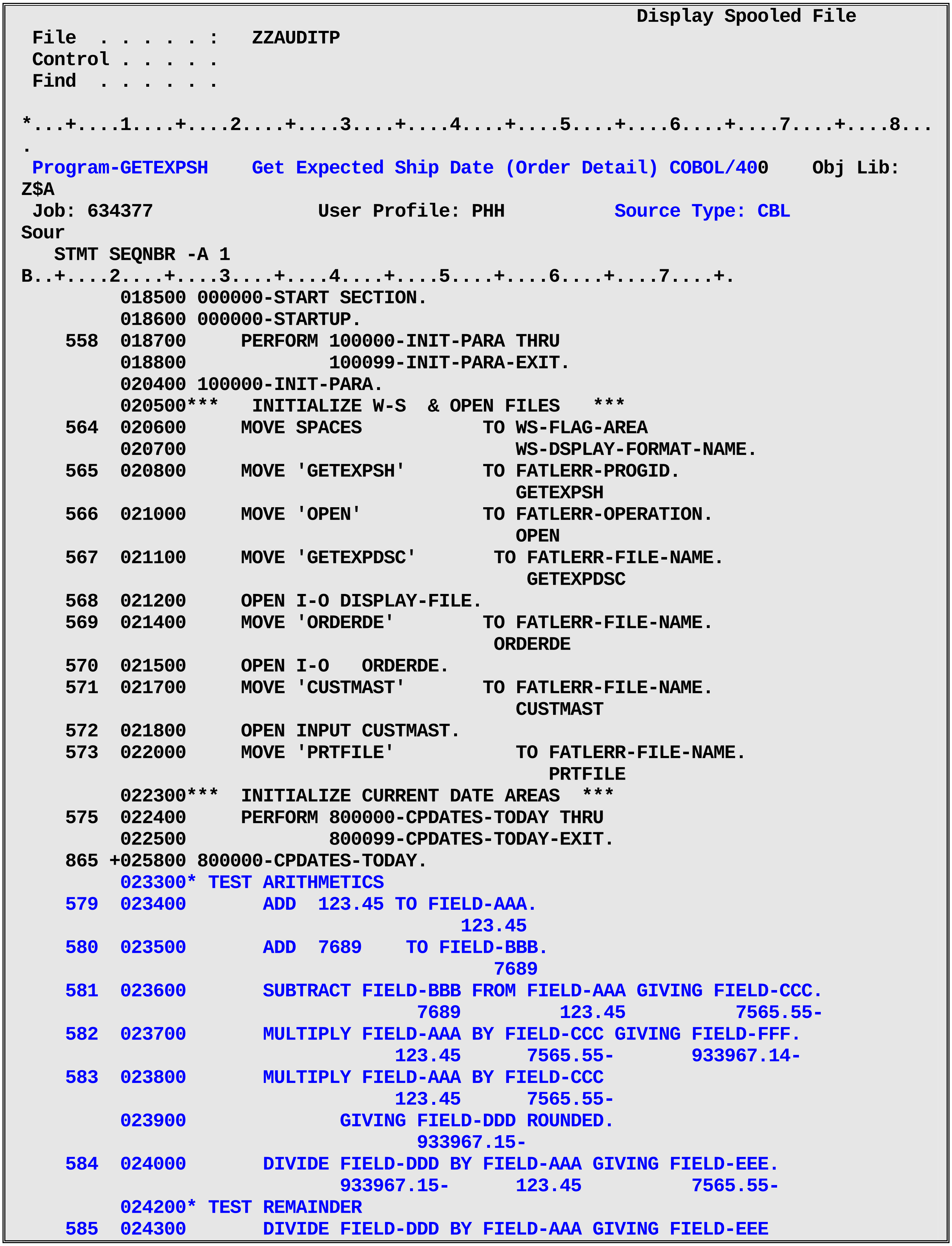
```
Z$PGM64R             Real-Time Program Audit Query (V4R3)              10/18/07
 PHH                User RTPA Query Summary Options Maintenance         12:14:22

 Type choices, press F5 to apply as User standard RTPA Query Summary options

  RTPA Query Summary Options:  Option

   Fields- Program Summary       Y    Y=Include

   Files- Program Summary        Y    Y=Include

   Fileio- Summary               Y    Y=Include

   Displays- Summary             Y    Y=Include

   Reports- Summary              Y    Y=Include

   Level- Program call level     Y    Y=Include




F3=Exit F5=Apply standard F12=Cancel (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure P.12 RTPA Query Summary Options

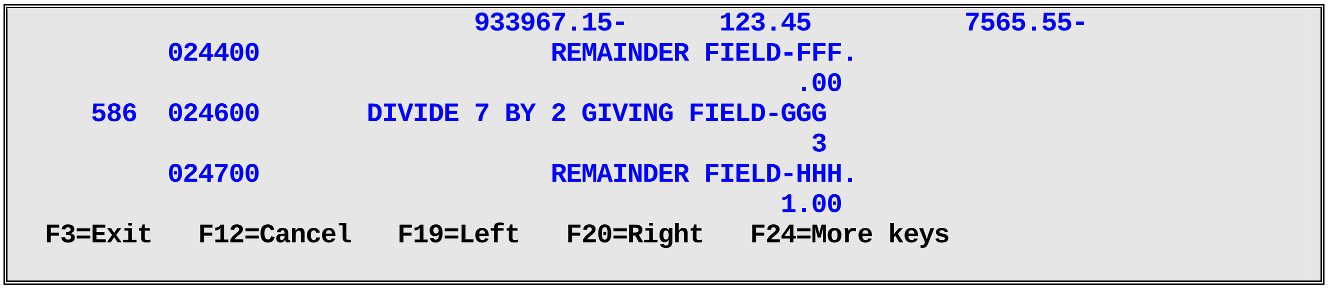


```
Z$COB01R           Real-Time Program Audit for COBOL (V4R3)      Date:  1/28/08
 PHH                          Detailed Job Record                 Time: 15:17:53

 Program   GETEXPSH   Get Expected Ship Date (Order Detail) COBOL/400
  Status   8 EXPAND CBL COMPILED OK

 Type options, press Enter.
   5=Display Compile listing   P=PDF Compile listing
 Opt Job     Job #   Records  Submitted          Completed              Elapsed
     Input   634477    1,098   1/28/08 14:54:30   1/28/08 14:54:34         4
     Insert  634479    3,812   1/28/08 14:54:35   1/28/08 14:54:43         8
     Expand  634482    4,910   1/28/08 14:54:45   1/28/08 14:54:49         4

 Source File QCBLSRC      Object Lib Z$AUDITE                Declaratives   Y
     Library Z$AUDIT      Audit JOBQ RTPA
     CBL Ver C CBL        Audit OUTQ
    From        To        Audit JOBD *LIBL
    From        To         JOBD Libr
    From        To
    From        To
    From        To
 F3=Exit                     F14=Fields     F15=Commands    F16=Variables
            F19=Called Pgm                                  F23=Pre-audit
                                     (C) 2000-2002 Harkins Audit Software, Inc.
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure P.13 RTPA for COBOL Expansion of COBOL program GETEXPSH

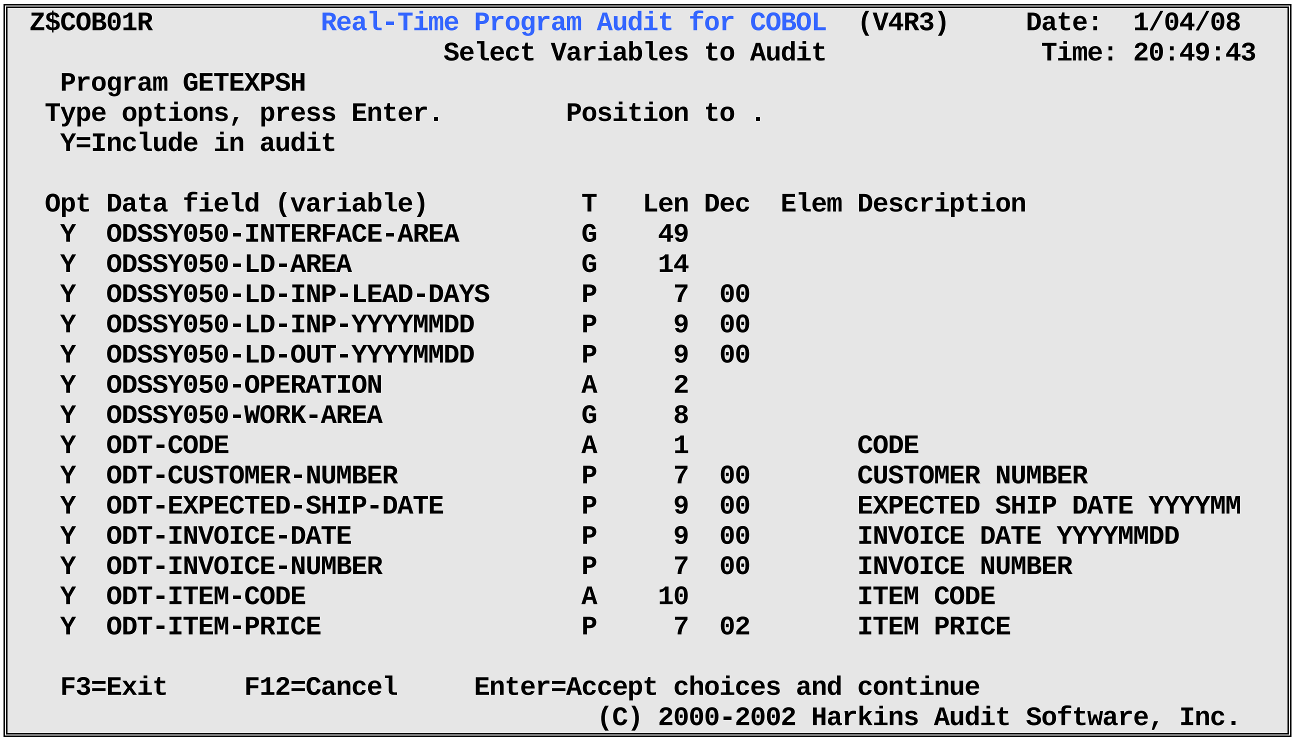


```
                                                        Display Spooled File
 File  . . . . . :   ZZAUDITP
 Control . . . . .
 Find  . . . . . .

*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8...
.
 Program-GETEXPSH    Get Expected Ship Date (Order Detail) COBOL/400    Obj Lib:
Z$A
 Job: 634377               User Profile: PHH          Source Type: CBL
Sour
   STMT SEQNBR -A 1
B..+....2....+....3....+....4....+....5....+....6....+....7....+.
         018500 000000-START SECTION.
         018600 000000-STARTUP.
    558  018700     PERFORM 100000-INIT-PARA THRU
         018800             100099-INIT-PARA-EXIT.
         020400 100000-INIT-PARA.
         020500***   INITIALIZE W-S  & OPEN FILES   ***
    564  020600     MOVE SPACES          TO WS-FLAG-AREA
         020700                             WS-DSPLAY-FORMAT-NAME.
    565  020800     MOVE 'GETEXPSH'      TO FATLERR-PROGID.
                                             GETEXPSH
    566  021000     MOVE 'OPEN'          TO FATLERR-OPERATION.
                                             OPEN
    567  021100     MOVE 'GETEXPDSC'      TO FATLERR-FILE-NAME.
                                              GETEXPDSC
    568  021200     OPEN I-O DISPLAY-FILE.
    569  021400     MOVE 'ORDERDE'       TO FATLERR-FILE-NAME.
                                           ORDERDE
    570  021500     OPEN I-O   ORDERDE.
    571  021700     MOVE 'CUSTMAST'      TO FATLERR-FILE-NAME.
                                             CUSTMAST
    572  021800     OPEN INPUT CUSTMAST.
    573  022000     MOVE 'PRTFILE'          TO FATLERR-FILE-NAME.
                                                PRTFILE
         022300***  INITIALIZE CURRENT DATE AREAS  ***
    575  022400     PERFORM 800000-CPDATES-TODAY THRU
         022500             800099-CPDATES-TODAY-EXIT.
    865 +025800 800000-CPDATES-TODAY.
         023300* TEST ARITHMETICS
    579  023400       ADD  123.45 TO FIELD-AAA.
                                        123.45
    580  023500       ADD  7689    TO FIELD-BBB.
                                           7689
    581  023600       SUBTRACT FIELD-BBB FROM FIELD-AAA GIVING FIELD-CCC.
                                    7689         123.45          7565.55-
    582  023700       MULTIPLY FIELD-AAA BY FIELD-CCC GIVING FIELD-FFF.
                                  123.45      7565.55-       933967.14-
    583  023800       MULTIPLY FIELD-AAA BY FIELD-CCC
                                  123.45      7565.55-
         023900              GIVING FIELD-DDD ROUNDED.
                                    933967.15-
    584  024000       DIVIDE FIELD-DDD BY FIELD-AAA GIVING FIELD-EEE.
                             933967.15-      123.45          7565.55-
         024200* TEST REMAINDER
    585  024300       DIVIDE FIELD-DDD BY FIELD-AAA GIVING FIELD-EEE
```



```
                            933967.15-      123.45          7565.55-
          024400                      REMAINDER FIELD-FFF.
                                                     .00
     586  024600        DIVIDE 7 BY 2 GIVING FIELD-GGG
                                                    3
          024700                      REMAINDER FIELD-HHH.
                                                   1.00
  F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure P.14 RTPA for COBOL Audit Output for program GETEXPSH



```
 Z$COB01R           Real-Time Program Audit for COBOL  (V4R3)     Date:  1/04/08
                                Select Variables to Audit           Time: 20:49:43
   Program GETEXPSH
  Type options, press Enter.        Position to .
   Y=Include in audit

  Opt Data field (variable)          T   Len Dec  Elem Description
   Y  ODSSY050-INTERFACE-AREA        G    49
   Y  ODSSY050-LD-AREA               G    14
   Y  ODSSY050-LD-INP-LEAD-DAYS      P     7  00
   Y  ODSSY050-LD-INP-YYYYMMDD       P     9  00
   Y  ODSSY050-LD-OUT-YYYYMMDD       P     9  00
   Y  ODSSY050-OPERATION             A     2
   Y  ODSSY050-WORK-AREA             G     8
   Y  ODT-CODE                       A     1       CODE
   Y  ODT-CUSTOMER-NUMBER            P     7  00       CUSTOMER NUMBER
   Y  ODT-EXPECTED-SHIP-DATE         P     9  00       EXPECTED SHIP DATE YYYYMM
   Y  ODT-INVOICE-DATE               P     9  00       INVOICE DATE YYYYMMDD
   Y  ODT-INVOICE-NUMBER             P     7  00       INVOICE NUMBER
   Y  ODT-ITEM-CODE                  A    10           ITEM CODE
   Y  ODT-ITEM-PRICE                 P     7  02       ITEM PRICE

   F3=Exit     F12=Cancel     Enter=Accept choices and continue
                                   (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure P.15 RTPA for COBOL Audit Variables for program GETEXPSH

Productivity Gains with RTPA

Using RTPA auditing can save almost any programmer substantial amounts of time in many different kinds of programming activities. Real-Time Program Auditing is a powerful technique, allowing programmers to more quickly learn legacy programs, find bugs in software, validate that new software functions correctly, enhance programs, and essentially eliminate guessing and speculation as to what happened or is happening. Auditing is an intuitive process, and RTPA is so easy to use that most programmers can be up and running with audit-enabled programs within minutes of installing RTPA. It is not uncommon for programmers to report that they solved a major problem the first day that they installed the software; a problem that may have been plaguing them for a while but couldn't be found using of the conventional debugging techniques.

Demystifying Legacy Programs

Many programming tasks require that a programmer first learn a legacy program before getting to the real work of enhancing, correcting or updating it. These existing programs can be very large, complex and entirely unfamiliar to the programmer, resulting in a time-consuming, often unpleasant effort, and the data being processed is often unknown. The problem of learning existing programs is particularly poignant when the enhancement is small – in these cases, most of the time and effort is expended on learning the program.

RTPA can reduce the time and effort of the learning process dramatically. Instead of printing out the program or compiling the program, and reading through the listing trying to guess where the execution flow goes, RTPA shows the actual execution flow caused by the data being processed. Programmers don't have to guess anymore or spend time looking at sections of code that are not part of the actual program flow.

Fixing Software Glitches

A large part of a programmer's job is finding and fixing bugs in software. All software potentially has bugs in it, many of which are easy to find and fix once they occur. However, the most frustrating and potentially expensive bugs are those that are intermittent or are hard to recreate in a test environment. Often there is too much code for a programmer to review closely in a reasonable timeframe. If the programmer isn't even certain of where the bug occurs, stepper debuggers may not be able to help because the programmer doesn't know where to put breakpoints. To complicate matters further, some bugs only occur under very particular input conditions that the programmer may not be able to replicate in a test environment, or in attempting to reconstruct the exact data previously processed.

Thankfully, RTPA is uniquely powerful for finding difficult software bugs, with no programmer intervention required. When a failure in a test or production environment can't be easily replicated in the lab or if it is infrequent enough to make stepper-type debugging impractical, auditing the program can be the best way to find and fix the program. An audit-enabled program shows everything that occurs before, during and after the bug. By searching for the audit file by execution time, data values or specific operations, a programmer can find and review the problem in complete detail, and track back to the source or cause of the problem.

Improving Quality and Reliability

As the demand for productivity increases and applications become more complex and critical, most programmers also feel increased demand for program quality. Many programmers have had to become experts in software testing and validation as part of their jobs. The testing process can be slow and difficult in many cases, particularly when testing software modules, such as validating the values of data variables in subroutines and procedures.

RTPA offers programmers a convenient way to speed up testing and improve overall program quality. The audit allows a programmer to validate a program's that a logic and outputs are correct without having to spend time writing output specification and "hello" statements. RTPA makes it easy to review a program's execution without having to set breakpoints or write test code. RTPA audits are automatically produced when the enabled program is run, allowing programmer or auditor review in real-time as the program executes or later from the audit output file.

Creating Web GUI Reports from program data and logic

Many Web graphical Business Intelligence (BI) products utilize SQL to access System i DB2 databases to produce charts and reports for top management review and action. Many of these reports are now developed and produced in RPG or COBOL programs and printed in text formats using the System i Printer File (PRTF) capability. RTPA is very useful in identifying and matching the often complex

computations and the DB2 databases and database fields used in producing the summarized data that is actually used on these management reports.
Thus, RTPA allows a user to simply enter a report value, such as $3,561,014.59 for Sales YTD, and search the RTPA audit report from the program (or programs) that created the report to find the source statement and the field (variable) with this value. The audit report than can be searched (backwards) to locate exactly the source statements and data of how the Sales YTD computation of 3,561,014.59 was created and the databases and database field names, and any work fields in the program were computed.

This RTPA auditing and search capability on data values allows not only the recreation of the report Sales YTD amount in the BI Web graph and Web report in SQL, but also all the drilldown totals and computations to allow the BI report to drilldown to the actual source of the Sales YTD amount, whether the drilldown is by division, product line, year or other program logic from the DB2 databases.

The current text (printer file) program file I/O may be embedded SQL, or native File I/O., in which case the native File I/O (for example Read, Write, Chain) could be converted to corresponding SQL statements in the BI report with the identified DB2 files and DB2 fields logic and computations explicitly identified in RTPA auditing.

Advantages of Auditing with RTPA

RTPA demystifies complex programs enhancing application and program design, development, maintenance, and enhancement. The full RTPA audit lists the exact program source statements that are actually executed as the computer executes them, regardless of the structure of the program, or called programs. Thus, the programmer or analyst does not have to understand the source program or guess what might happen if different conditions or data is encountered.

RTPA electronic program auditing is, by far, the fastest and most productive and effective method developing, maintaining and supporting corporate applications over any debugging tool or known other analysis technique. No prior knowledge of the application, program, files, data other details is required. Simply enable the application programs with RTPA capability, run the application, and observe exactly how the computer is actually executing or actually executed the application programs and the data using the audit disk or printed output.

Hands-Free Operation. Once you audit-enable a program with RTPA, you don't have to set program breakpoints or stop the normal flow of the program as it executes. And, the programmer does not have to be present when the program executes.

Video camera like complete auditing to disk. RTPA for RPG, CLP, and COBOL default to completely audit every executing program statement in RTPA enabled programs to a spool file, including the source statement, contents of variables, and the exact moment the statement was executed. The RTPA Query product combines all of the called programs in a job, such as Order maintenance, or Customer invoicing, together by the moment the program was executed by the computer.

Flexible Auditing Choices. You can make your audit as in-depth or as high-level as you like. An audit can include every executable source statement in the source program and the value of every variable, or it can focus on selected program files, records, operations, variables, and subroutines. Audit output may also be started, stopped, and resumed based on the data contents of variables, or the relationships among variables in the executing program.

Low System Overhead. RTPA has a very small impact on overall system performance, allowing you to test and validate your software while it runs at near-normal speeds. In real-world tests, auditing a program adds as little as 10% to total CPU time to a job on an System i computer.

What RTPA Doesn't Do

We would love to be able to say that RTPA is the only tool that a programmer needs to become more productive and solve every problem. However, RTPA does have some limitations which you should be aware of.

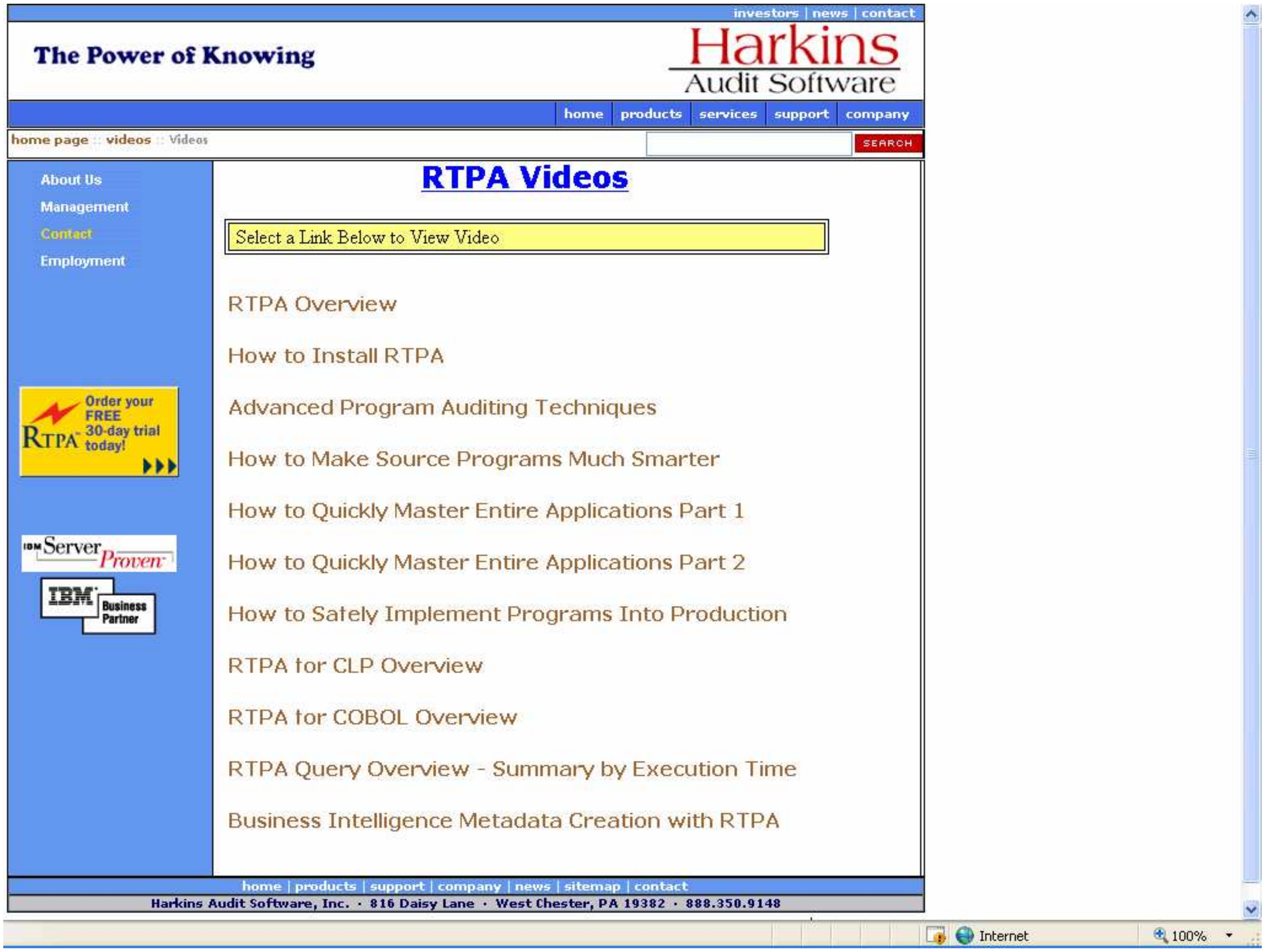
RTPA is intended for use in a programming development and testing environment. It is not intended for (and we don't allow) inclusion in shrink-wrapped code or any code that is sold to another party.

Document Conventions

- `courier text` is used to indicate a literal statement
- *Italic text* is used to indicate a variable
- F*N* is used to denote function (or command key) number

RTPA Web video presentations for programmer orientation

RTPA for RPG Camtasia videos PC presentations are available online at www.harkinsaudit.com for RTPA education and programmer orientation. These Camtasia video presentations supplement the information in this Users Manual.

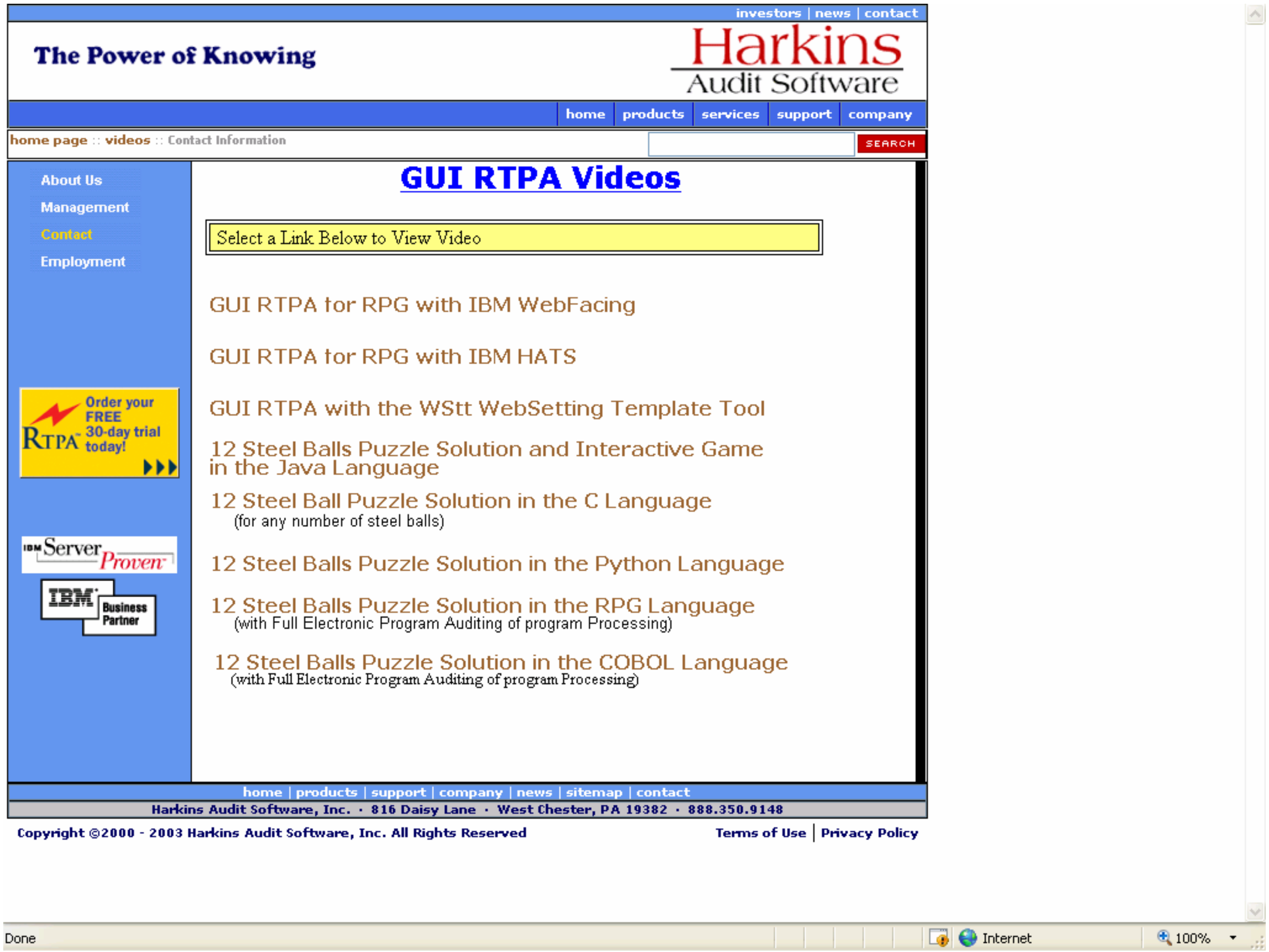


RTPA Videos

Select a Link Below to View Video

- RTPA Overview
- How to Install RTPA
- Advanced Program Auditing Techniques
- How to Make Source Programs Much Smarter
- How to Quickly Master Entire Applications Part 1
- How to Quickly Master Entire Applications Part 2
- How to Safely Implement Programs Into Production
- RTPA for CLP Overview
- RTPA for COBOL Overview
- RTPA Query Overview - Summary by Execution Time
- Business Intelligence Metadata Creation with RTPA

Harkins Audit Software, Inc. · 816 Daisy Lane · West Chester, PA 19382 · 888.350.9148



GUI RTPA Videos

Select a Link Below to View Video

GUI RTPA for RPG with IBM WebFacing

GUI RTPA for RPG with IBM HATS

GUI RTPA with the WStt WebSetting Template Tool

12 Steel Balls Puzzle Solution and Interactive Game in the Java Language

12 Steel Ball Puzzle Solution in the C Language
(for any number of steel balls)

12 Steel Balls Puzzle Solution in the Python Language

12 Steel Balls Puzzle Solution in the RPG Language
(with Full Electronic Program Auditing of program Processing)

12 Steel Balls Puzzle Solution in the COBOL Language
(with Full Electronic Program Auditing of program Processing)

Harkins Audit Software, Inc. · 816 Daisy Lane · West Chester, PA 19382 · 888.350.9148

Copyright ©2000 - 2003 Harkins Audit Software, Inc. All Rights Reserved

Terms of Use | Privacy Policy

RTPA Online Demonstrations and Training

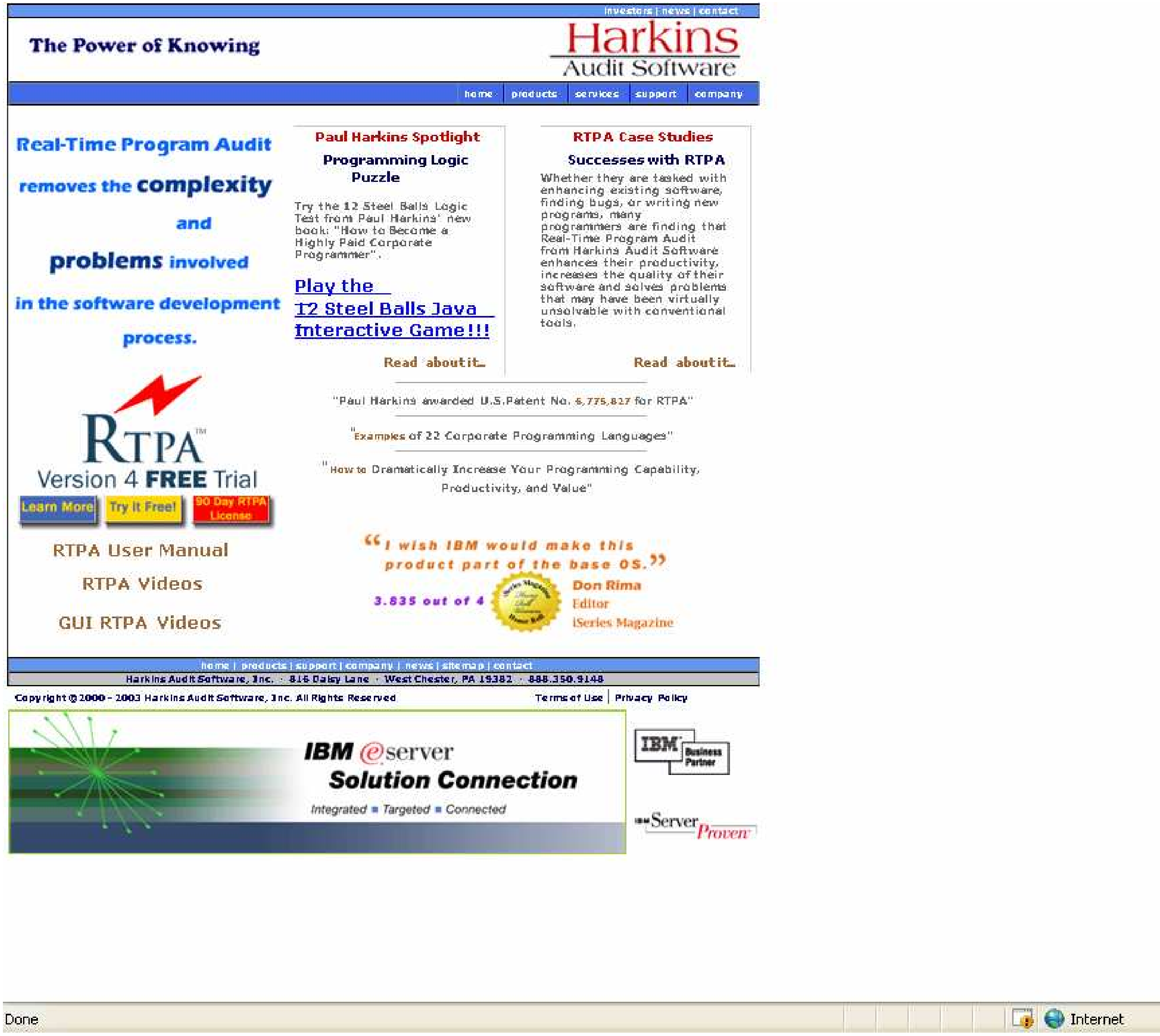
Online and interactive RTPA demonstrations and training are available worldwide at the user location via Microsoft Net Meeting. The user requirements are a Windows PC and a high-speed Internet connection.

RTPA Software Guarantee

RTPA Software is available "AS IS' without warrantee expressed or implied.

Harkins Audit Software, Inc. Website

Harkins Audit Software, Inc. develops, maintains, and supports the RTPA family of programmer productivity software products. Web site www.harkinsaudit.com contains information about these products, downloadable program code, videos, manuals, and other information about these products.



investors | news | contact

The Power of Knowing

Harkins Audit Software

home | products | services | support | company

Real-Time Program Audit removes the complexity and problems involved in the software development process.

Paul Harkins Spotlight

Programming Logic Puzzle

Try the 12 Steel Balls Logic Test from Paul Harkins' new book: "How to Become a Highly Paid Corporate Programmer".

Play the 12 Steel Balls Java Interactive Game!!!

Read about it...

RTPA Case Studies

Successes with RTPA

Whether they are tasked with enhancing existing software, finding bugs, or writing new programs, many programmers are finding that Real-Time Program Audit from Harkins Audit Software enhances their productivity, increases the quality of their software and solves problems that may have been virtually unsolvable with conventional tools.

Read about it...

RTPA™ Version 4 FREE Trial

Learn More | Try It Free! | 90 Day RTPA License

"Paul Harkins awarded U.S.Patent No. 6,775,827 for RTPA"

"Examples of 22 Corporate Programming Languages"

"How to Dramatically Increase Your Programming Capability, Productivity, and Value"

RTPA User Manual

RTPA Videos

GUI RTPA Videos

"I wish IBM would make this product part of the base OS."

3.835 out of 4

Don Rima
Editor
iSeries Magazine

home | products | support | company | news | sitemap | contact

Harkins Audit Software, Inc. · 816 Daisy Lane · West Chester, PA 19382 · 888.350.9148

Copyright © 2000 - 2003 Harkins Audit Software, Inc. All Rights Reserved

Terms of Use | Privacy Policy

IBM eserver Solution Connection

Integrated ■ Targeted ■ Connected

IBM Business Partner

Server Proven

Done

Internet

Chapter

1

Chapter 1: **Installing Real-Time Program Audit**

This chapter outlines the steps for installing RTPA for the first time or as a complete reinstall for an upgrade release.

Requirements

IBM System i (AS/400) Model 170 (RISC) with RPGIII or RPGIV
OS/400 V5R3 or later
RTPA for RPG LODRUN software CD
QSECOFR password
About 80 megabytes of disk (for executable and sample programs)

Step 1A: Installing RTPA from CD

If you are installing RTPA from an optical disk (CD), please follow these instructions. If you are installing RTPA from a downloaded file, please skip to the next section, **Step 1B: Installing RTPA from a Downloaded File**.

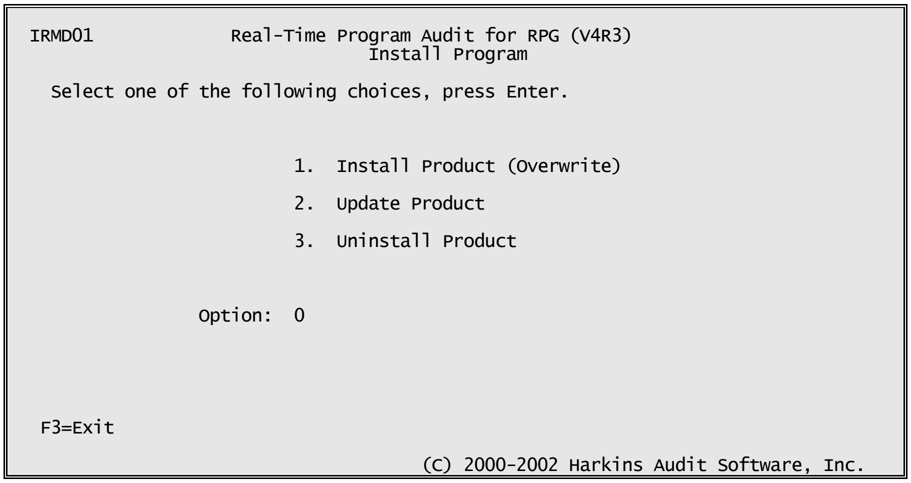
- ❖ Sign on as **QSECOFR**.

- ❖ Put the RTPA for RPG LODRUN CD in the iSeries (AS/400) CD reader.

- ❖ Enter the following command at the command line to start the installation process:

```
LODRUN *OPT
```

The following screen will be displayed:

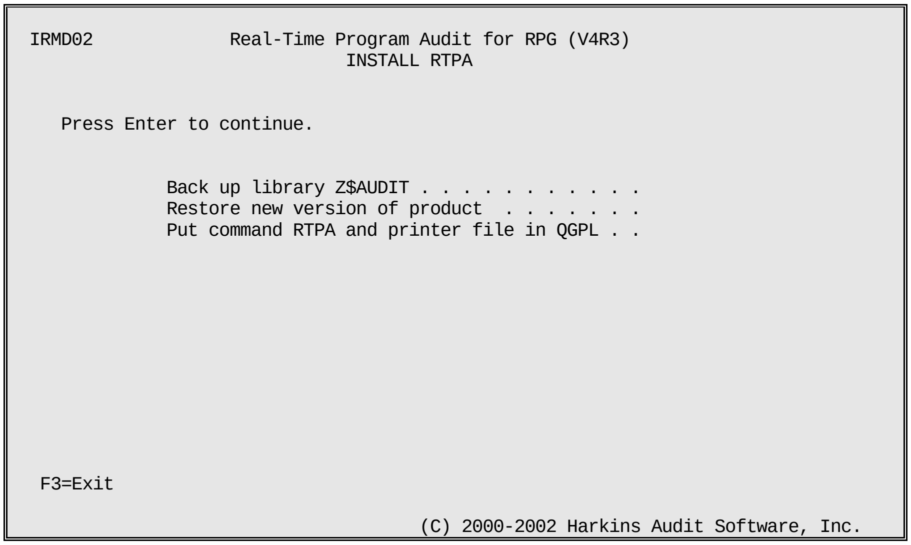


```
IRMD01            Real-Time Program Audit for RPG (V4R3)
                                Install Program

  Select one of the following choices, press Enter.


                        1.  Install Product (Overwrite)

                        2.  Update Product

                        3.  Uninstall Product


               Option:  0




 F3=Exit
                                       (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 1.1 RTPA LODRUN Screen

- Choose **option 1** to install RTPA.
- At the next screen press **Enter** while RTPA copies itself onto your computer.

When the procedure is complete, you will see the following screen:



```
IRMD02            Real-Time Program Audit for RPG (V4R3)
                              INSTALL RTPA


   Press Enter to continue.


             Back up library Z$AUDIT . . . . . . . . . . .
             Restore new version of product  . . . . . . .
             Put command RTPA and printer file in QGPL . .










 F3=Exit
                                       (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 1.2 RTPA LODRUN Installation Successful

Step 1B: Installing RTPA from a Downloaded File

If you are installing RTPA from a downloaded file, please follow these instructions. If you are installing RTPA from an optical disk (CD), please skip to the previous section, **Step 1A: Installing RTPA from CD**.

The RTPA video **How to Install RTPA** illustrates these steps on www.harkinsaudit.com.

- Download the files from the Harkins Audit Website. www.harkinsaudit.com

  - From your PC, go to http://www.harkinsaudit.com/download/rtpa.shtml, or click on the 30-day trial button and follow the steps to download RTPA for RPG to your PC.
  - Download the file RTPA40B1.ZIP to your PC into directory such as `C:\temp`
    The Folder **temp** in local drive C of your PC maybe created in the download process.
    This download of the RTPA for RPG licensed code to your PC should take less than two minutes (see the How to Install RTPA video).
  - WINZIP (Unzip) the downloaded RTPA40B1.ZIP savf on the PC to expand the file into a directory such as C:\temp, to extract the .SAVF file
  - The WINZIP (ZIP and UNZIP files) is available on the Internet.

```
ftp>
```

- On the AS/400, create a save file.

```
CRTSAVF FILE(QGPL/RTPA40B1)
```

- FTP from the PC to the save file on the AS/400 (System i).

  - From a DOS command prompt, type: (Start, Run, Cmd for DOS window)

```
ftp XXX.XXX.XXX.XXX
```

where XXX.XXX.XXX.XXX is the IP address of the AS/400.

  - Provide the username QSECOFR and the proper password.
  - Transfer the file by typing:

```
lcd c:\temp
bin
cd qgpl
put rtpa40b1.savf
quit
```

The FTP of the PC file to the AS/400 should take about ten seconds.

- Restore the object RTPA40B1. From the AS/400, type

```
RSTOBJ OBJ(*ALL) SAVLIB(QGPL) DEV(*SAVF) SAVF(RTPA40B1)
```

- Run the installation program. From the AS/400, type

```
CALL QGPL/IRM
```

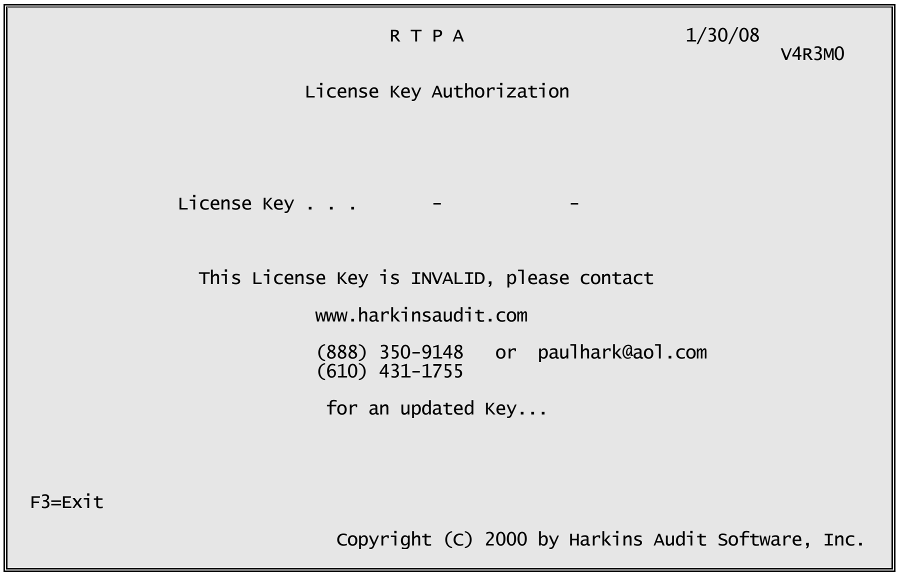
Step 2: Enter the RTPA for RPG License Key

- Please Email paulhark@aol.com for a 30 Day free RTPA license Key

- From the command line, type:

```
RTPA
```

- Tab to the **`License Key`** area. Enter the license key and press **Enter**.



```
                             R T P A                  1/30/08
                                                             V4R3M0

                     License Key Authorization



            License Key . . .      -            -


             This License Key is INVALID, please contact

                        www.harkinsaudit.com

                        (888) 350-9148   or  paulhark@aol.com
                        (610) 431-1755

                         for an updated Key...



F3=Exit

                          Copyright (C) 2000 by Harkins Audit Software, Inc.
```

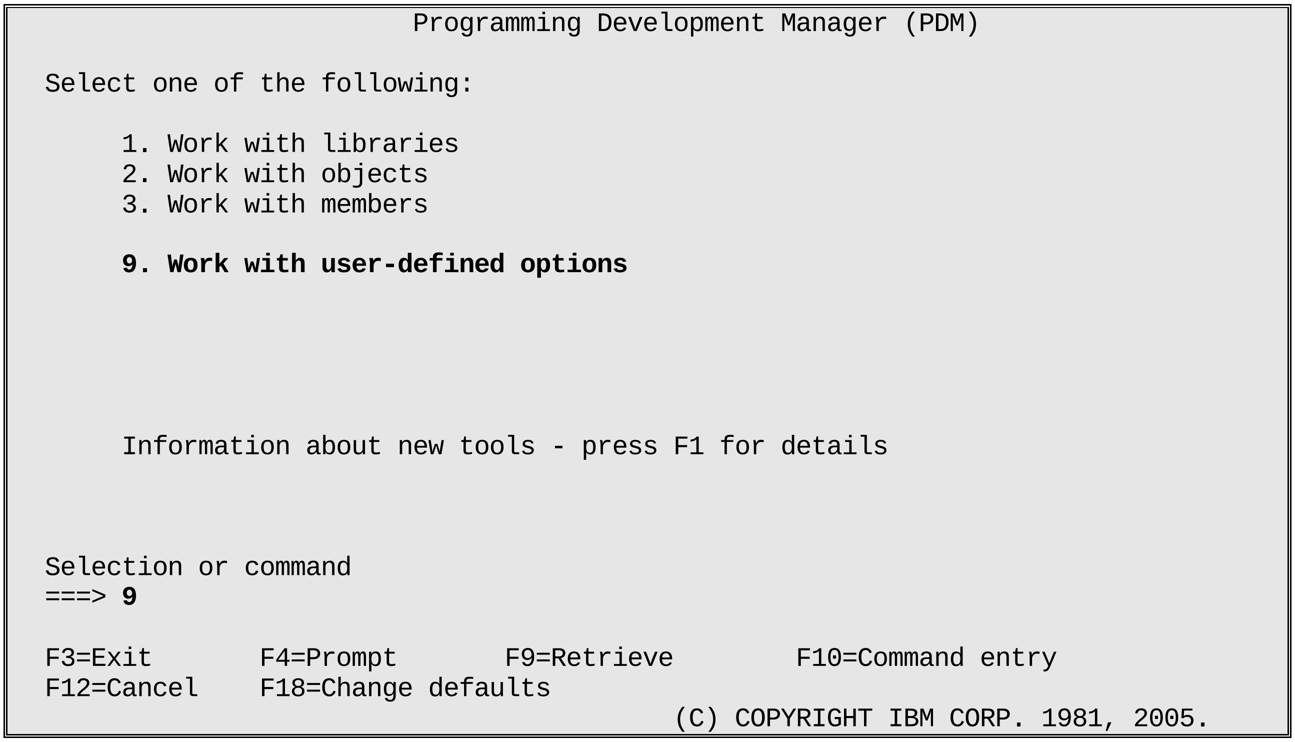
Figure 1.3 RTPA License Key Authorization Screen

The software is now loaded on the iSeries computer and ready to use.

Note: In some cases, the software may already have a key loaded as part of the distribution package. In these cases, you will not see the License Key Authorization screen. This screen will automatically appear when an evaluation license expires. You may contact the sales department to obtain evaluation license keys at any time, without having to reload the software.

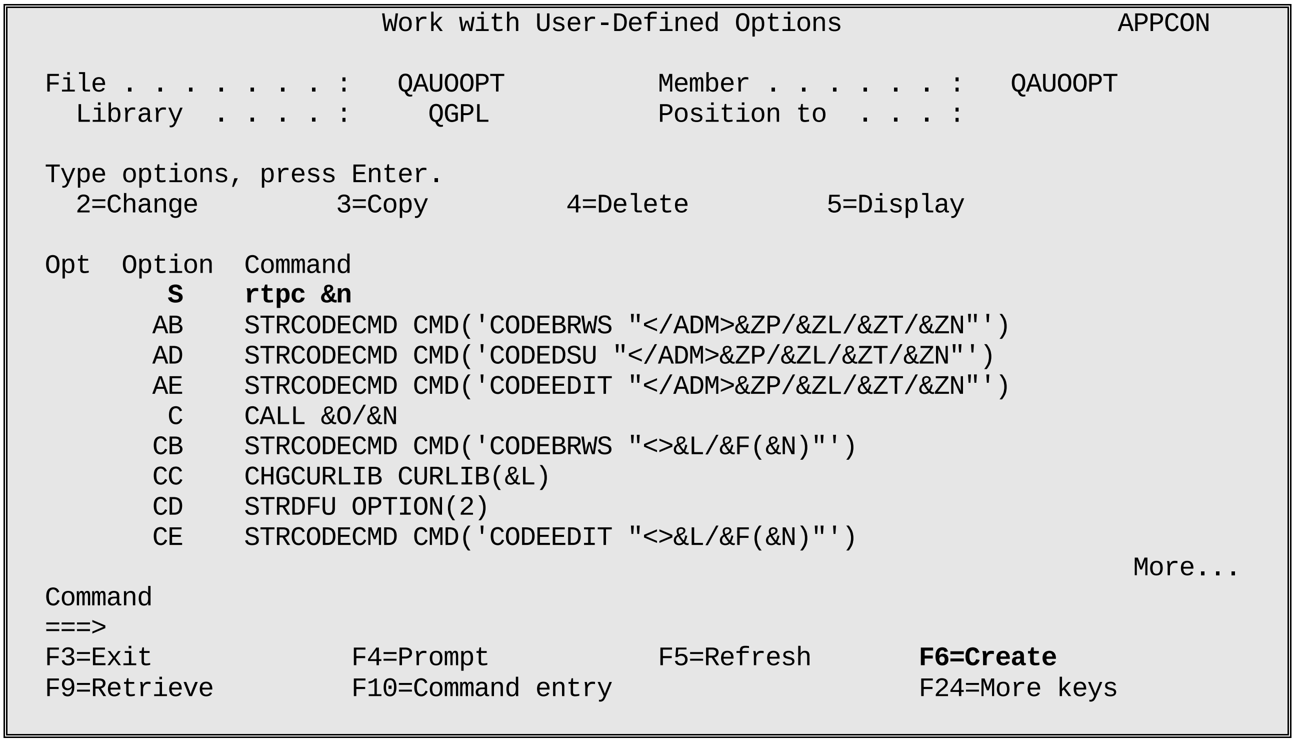
How to create a PDM User-defined Option for RTPA

A final RTPA installation step may be optionally accomplished to define a PDM User-defined option to allow PDM to pass a selected PDM source program name back into the RTPA Main screen when PDM is selected in the RTPA main screen.



```
                           Programming Development Manager (PDM)

  Select one of the following:

       1. Work with libraries
       2. Work with objects
       3. Work with members

       9. Work with user-defined options




       Information about new tools - press F1 for details



  Selection or command
  ===> 9

  F3=Exit       F4=Prompt       F9=Retrieve        F10=Command entry
  F12=Cancel    F18=Change defaults
                                          (C) COPYRIGHT IBM CORP. 1981, 2005.
```

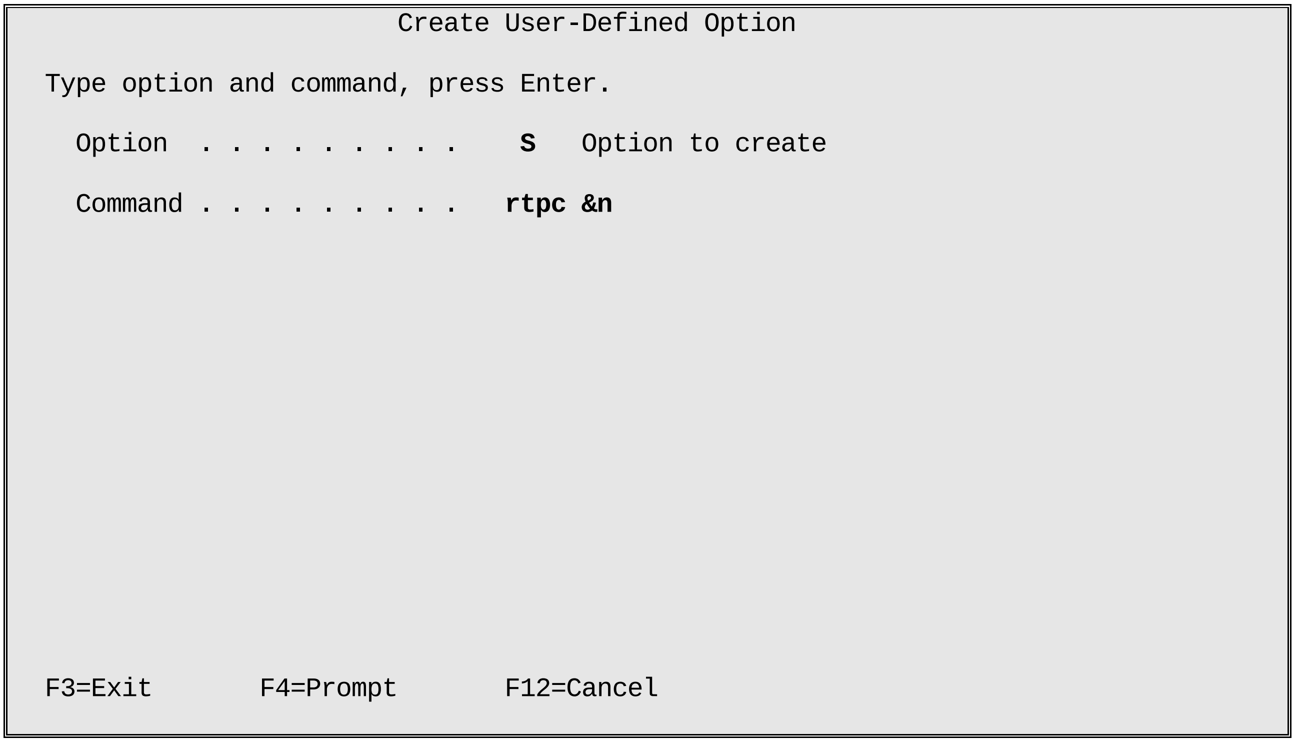
Figure 1.4 Select PDM option 9 to create a User-defined PDM Option



```
                         Work with User-Defined Options                  APPCON

  File . . . . . . . :   QAUOOPT          Member . . . . . . :   QAUOOPT
    Library  . . . . :     QGPL           Position to  . . . :

  Type options, press Enter.
    2=Change         3=Copy         4=Delete         5=Display

  Opt  Option  Command
          S    rtpc &n
         AB    STRCODECMD CMD('CODEBRWS "</ADM>&ZP/&ZL/&ZT/&ZN"')
         AD    STRCODECMD CMD('CODEDSU "</ADM>&ZP/&ZL/&ZT/&ZN"')
         AE    STRCODECMD CMD('CODEEDIT "</ADM>&ZP/&ZL/&ZT/&ZN"')
          C    CALL &O/&N
         CB    STRCODECMD CMD('CODEBRWS "<>&L/&F(&N)"')
         CC    CHGCURLIB CURLIB(&L)
         CD    STRDFU OPTION(2)
         CE    STRCODECMD CMD('CODEEDIT "<>&L/&F(&N)"')
                                                                        More...
  Command
  ===>
  F3=Exit             F4=Prompt           F5=Refresh        F6=Create
  F9=Retrieve         F10=Command entry                     F24=More keys
```

Figure 1.5 Press F6 to create the RTPA PDM User-defined option

Note – The RTPA User-defined option S is already in the list of PDM user-defined options



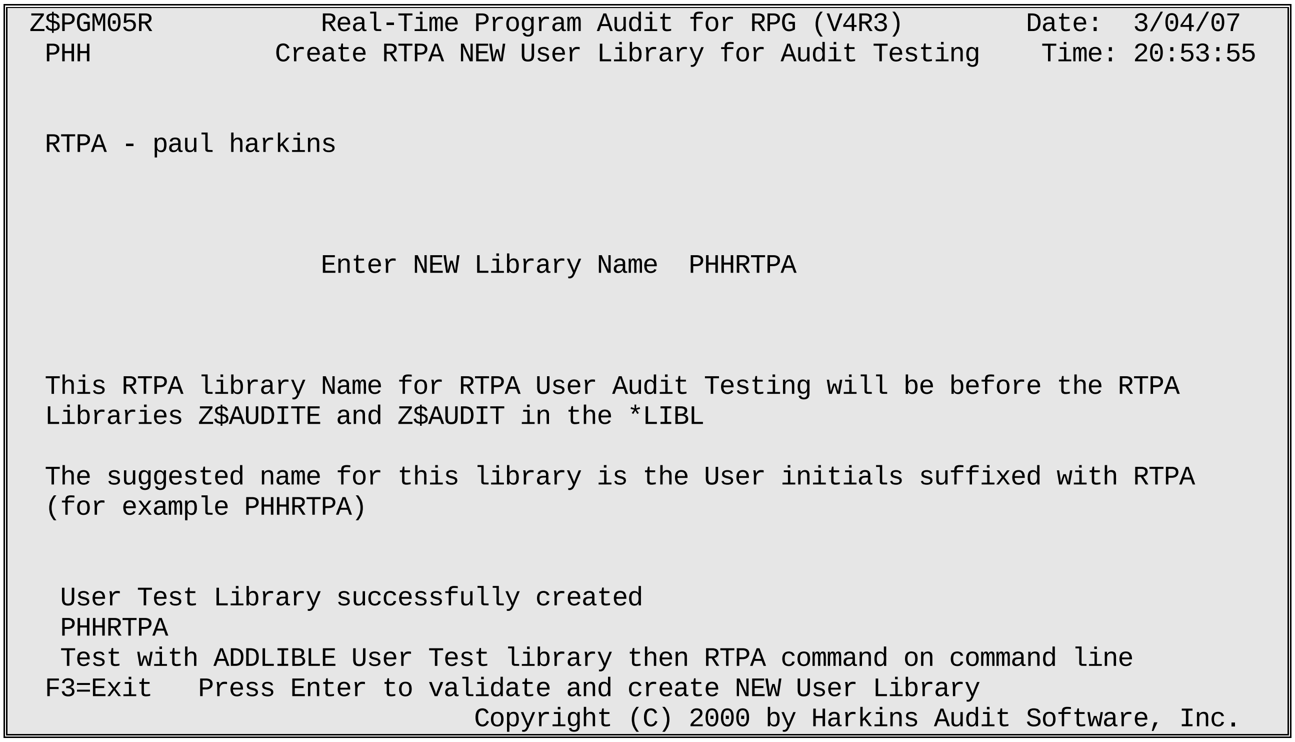
```
                         Create User-Defined Option

 Type option and command, press Enter.

   Option  . . . . . . . . .    S   Option to create

   Command . . . . . . . . .   rtpc &n










 F3=Exit       F4=Prompt       F12=Cancel
```

Figure 1.6 Enter the PDM RTPA User-defined option to allow selection of a member name in PDM

This PDM RTPA User-defined option allows PDM to pass a member name selected with an S (blank then S and the Enter Key) back to the RTPA main screen in the source member name field, when PDM is selected with command key 4 and the cursor is on the source member name field in the RTPA main screen.

How to create a private RTPA User Testing Library

Each programmer using RTPA may create a private testing library by selecting Option 6 of the RTPA Maintenance menu (F9 on the RTPA main screen). This library may be placed first in the library list with the ADDLIBLE command.

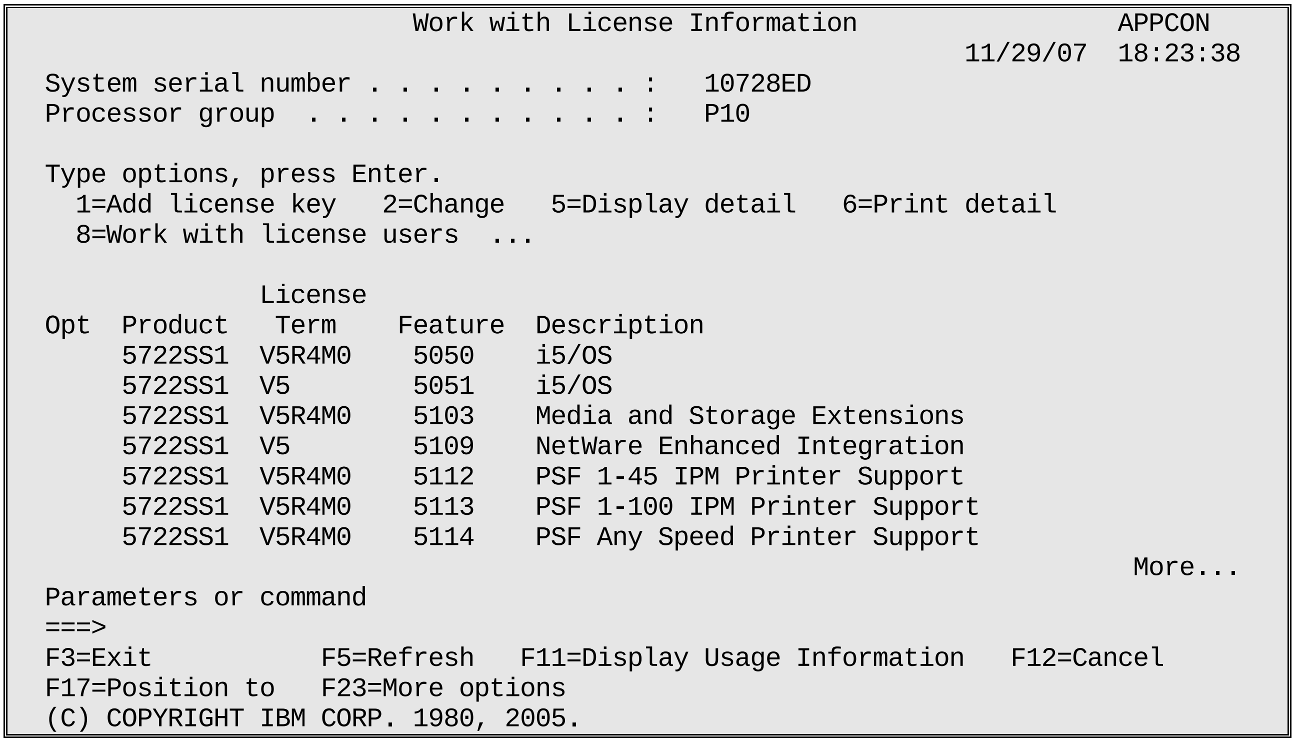


```
Z$PGM05R           Real-Time Program Audit for RPG (V4R3)       Date:  3/04/07
 PHH           Create RTPA NEW User Library for Audit Testing    Time: 20:53:55


 RTPA - paul harkins



                  Enter NEW Library Name  PHHRTPA



 This RTPA library Name for RTPA User Audit Testing will be before the RTPA
 Libraries Z$AUDITE and Z$AUDIT in the *LIBL

 The suggested name for this library is the User initials suffixed with RTPA
 (for example PHHRTPA)


  User Test Library successfully created
  PHHRTPA
  Test with ADDLIBLE User Test library then RTPA command on command line
 F3=Exit   Press Enter to validate and create NEW User Library
                             Copyright (C) 2000 by Harkins Audit Software, Inc.
```

Figure 1.7 Creation of a private User testing library named PHHRTPA

How to find the System i Processor Group with WRKLICINF

Use the WRKLICINF command to display the System i Processor Group, serial number, and installed IBM software with feature codes.
Use the DSPSYSVAL command with system value QMODEL to display the System i model number..



```
                          Work with License Information                APPCON
                                                              11/29/07  18:23:38
System serial number . . . . . . . . . :   10728ED
Processor group  . . . . . . . . . . . :   P10

Type options, press Enter.
  1=Add license key   2=Change   5=Display detail   6=Print detail
  8=Work with license users  ...

              License
Opt  Product   Term    Feature  Description
     5722SS1  V5R4M0    5050    i5/OS
     5722SS1  V5        5051    i5/OS
     5722SS1  V5R4M0    5103    Media and Storage Extensions
     5722SS1  V5        5109    NetWare Enhanced Integration
     5722SS1  V5R4M0    5112    PSF 1-45 IPM Printer Support
     5722SS1  V5R4M0    5113    PSF 1-100 IPM Printer Support
     5722SS1  V5R4M0    5114    PSF Any Speed Printer Support
                                                                       More...
Parameters or command
===>
F3=Exit           F5=Refresh   F11=Display Usage Information   F12=Cancel
F17=Position to   F23=More options
(C) COPYRIGHT IBM CORP. 1980, 2005.
```

Figure 1.8 Find the System i Processor Group and Serial number with WRKLICINF

Chapter

2

Chapter 2: **Quick Start Guide**

This chapter gives you a simple demonstration of RTPA with a sample program to help you get started with RTPA for the first time.

In this chapter, we will:

- Expand an RPGIV program (GETEXPSH)
- Compile and execute the expanded program
- Review the audit file to learn how the program works

Expand the Sample Program

❖ Launch RTPA by typing at the command line:

```
RTPA
```

❖ Select **NEWEXPSH** for expansion. This is an RPGLE source member in source file QRPGLESRC in library Z$AUDIT.

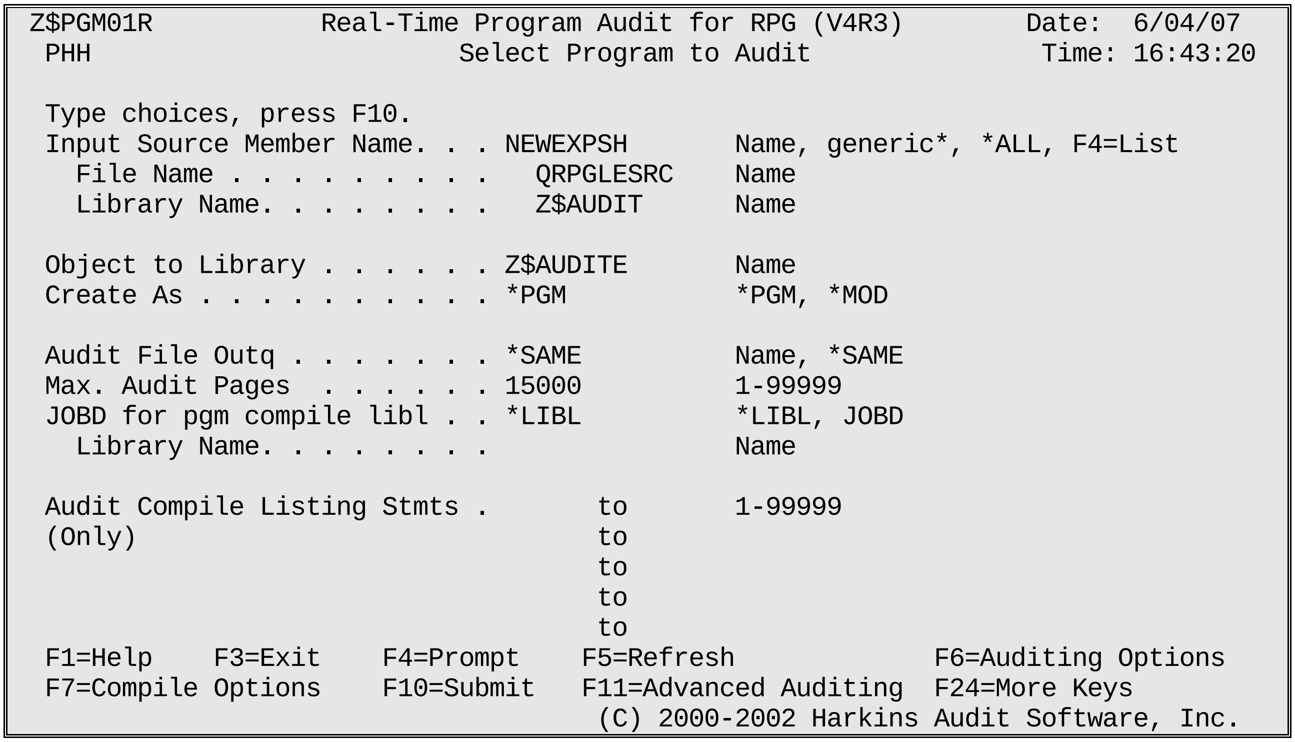
❖ Select an appropriate library in which to put the audit-enabled object program (such as **Z$AUDITE**). The expanded RTPA enabled object program can be placed in any library, but should not overlay the production object.

❖ Select an output queue for the RTPA audit output file ZZAUDITP.

❖ Select the Job Description and the Job Description Library that contains the libraries needed for the source program files used. The default JOBD and JOBD library for the User Profile is displayed from the signed on User Profile.

❖ The RTPA sample programs use the JOBD RTPA and JOBD Library QGPL.

Tip: Create a library exclusively for your audit-enabled program objects so you can always remember where they are. By adding and removing that library from your library list, you can change from executing the expanded object to executing the normal object. Or, you can use the RTPA provided Z$AUDITE library (RTPA Expanded library).



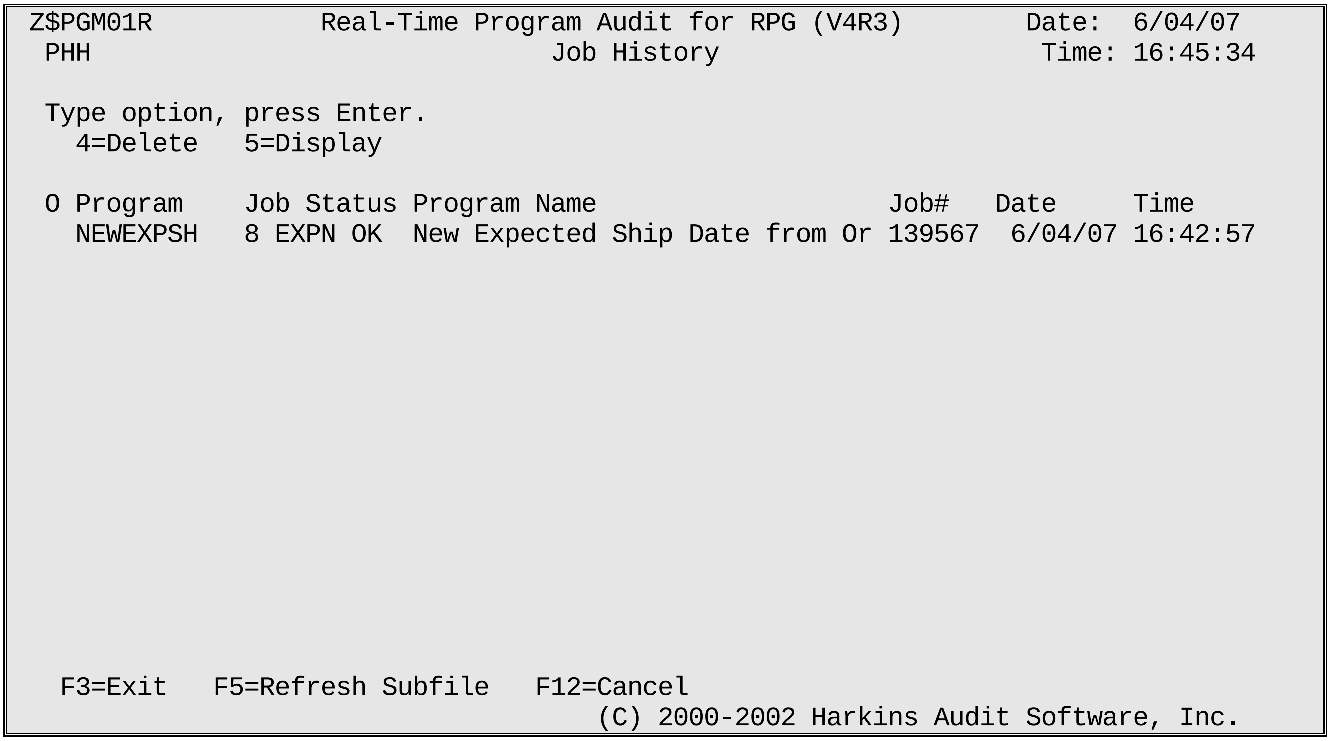
```
Z$PGM01R          Real-Time Program Audit for RPG (V4R3)       Date:  6/04/07
 PHH                         Select Program to Audit             Time: 16:43:20

 Type choices, press F10.
 Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QRPGLESRC    Name
   Library Name. . . . . . . .   Z$AUDIT      Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 15000          1-99999
 JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
   Library Name. . . . . . . .                Name

 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F1=Help    F3=Exit    F4=Prompt    F5=Refresh             F6=Auditing Options
 F7=Compile Options    F10=Submit   F11=Advanced Auditing  F24=More Keys
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 2.1 Select Program to Audit Screen with NEWEXPSH Selected

- Next, expand and compile the program by pressing **F10**. This is the normal expansion which audits virtually every executable source statement and all the data being processed.

The message: **Member NEWEXPSH   submitted. Press F18 to see status**. is displayed at the bottom of the screen, and the input source program goes through eight status steps to audit enable the expanded object program.

- To verify that the expansion was successful, press **F18** to get the screen shown in Figure 2.2.



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  6/04/07
 PHH                              Job History                     Time: 16:45:34

 Type option, press Enter.
   4=Delete   5=Display

 O Program    Job Status Program Name                  Job#   Date     Time
   NEWEXPSH   8 EXPN OK  New Expected Ship Date from Or 139567  6/04/07 16:42:57










  F3=Exit   F5=Refresh Subfile   F12=Cancel
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

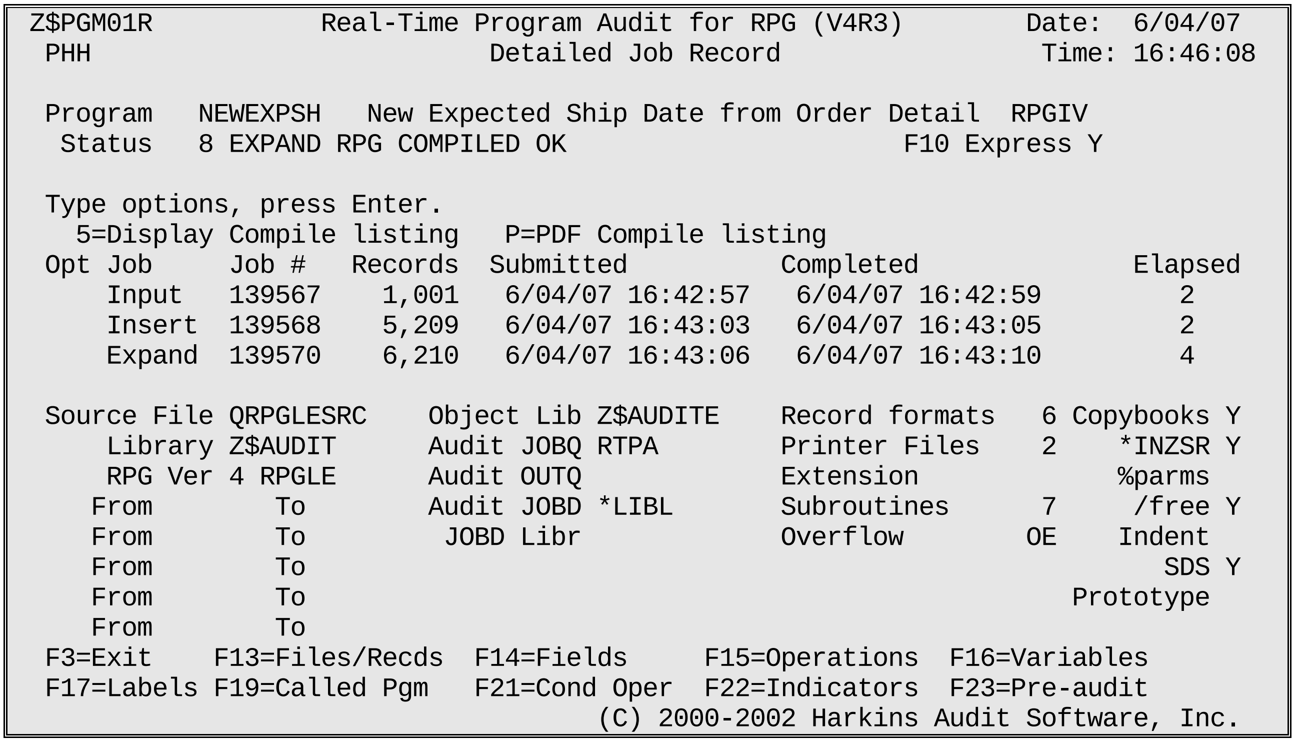
Figure 2.2 Programmer Audit Compiles by submitted Job# Screen (for Today)

In Figure 2.2, the status of 8 EXPN OK indicates that the source code was expanded and that the compile of the expanded program completed successfully. If the program has not reached status code 8, you may press Enter to refresh the screen. This screen shows all RTPA expanded programs (Jobs) for Today.

Status 8 Expand OK means that the RTPA audit enabled object program (from the audit enabled source copy of the program in Z$AUDITE) has been successfully created, and the program may be run to produce an RTPA audit.

- Exit RTPA by pressing **F3** twice.

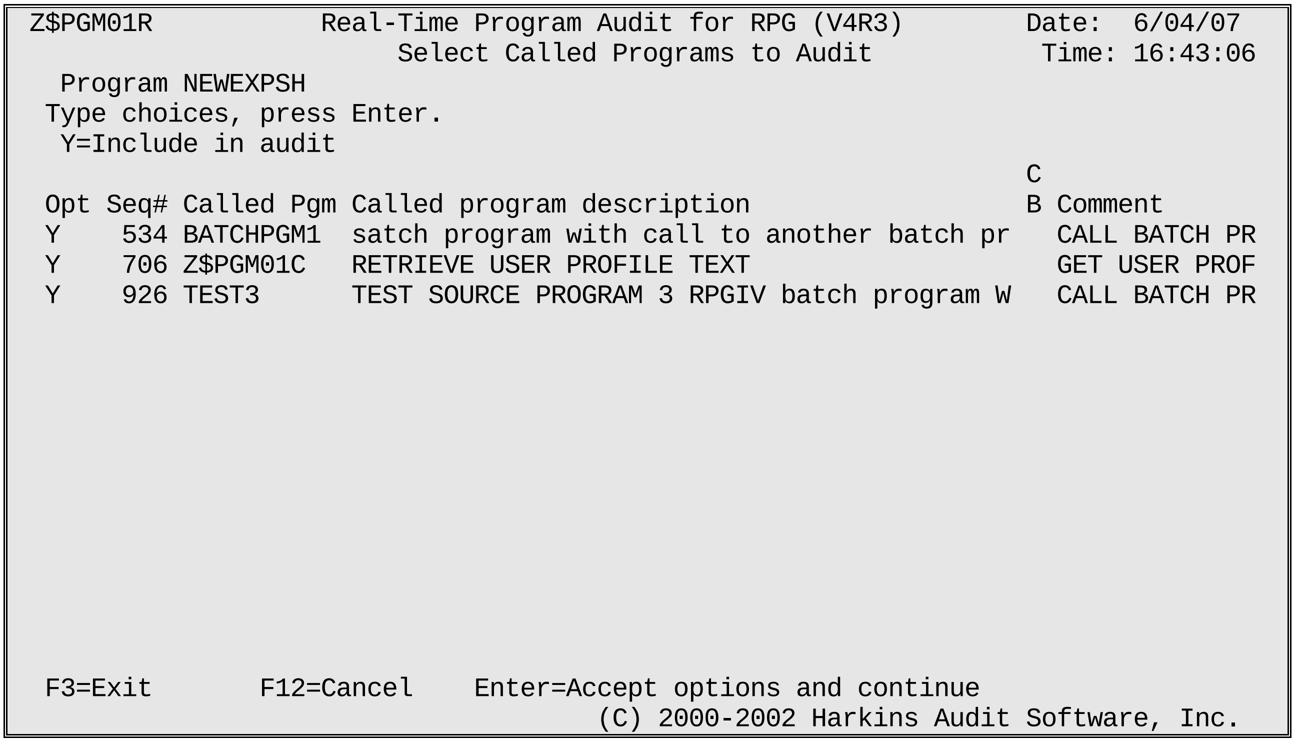
Note – Entering a 5 to the left of the program name displays the RTPA detailed Job History screen, which is informational only.



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)       Date:  6/04/07
 PHH                            Detailed Job Record                Time: 16:46:08

 Program   NEWEXPSH   New Expected Ship Date from Order Detail  RPGIV
  Status   8 EXPAND RPG COMPILED OK                      F10 Express Y

 Type options, press Enter.
   5=Display Compile listing   P=PDF Compile listing
 Opt Job     Job #   Records  Submitted          Completed             Elapsed
     Input   139567    1,001   6/04/07 16:42:57   6/04/07 16:42:59         2
     Insert  139568    5,209   6/04/07 16:43:03   6/04/07 16:43:05         2
     Expand  139570    6,210   6/04/07 16:43:06   6/04/07 16:43:10         4

 Source File QRPGLESRC    Object Lib Z$AUDITE    Record formats   6 Copybooks Y
     Library Z$AUDIT      Audit JOBQ RTPA        Printer Files    2    *INZSR Y
     RPG Ver 4 RPGLE      Audit OUTQ             Extension             %parms
    From        To        Audit JOBD *LIBL       Subroutines      7     /free Y
    From        To         JOBD Libr             Overflow        OE    Indent
    From        To                                                        SDS Y
    From        To                                                  Prototype
    From        To
 F3=Exit    F13=Files/Recds  F14=Fields     F15=Operations  F16=Variables
 F17=Labels F19=Called Pgm   F21=Cond Oper  F22=Indicators  F23=Pre-audit
                                    (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 2.3 RTPA Expanded Job Detailed Information screen

- Pressing Command Key 19 displays all called programs from this expanded program.
- Expanding these called programs will provide RTPA program audits of these called programs when the NEWEXPSH expanded object program is executed.
- Expanding all the application source programs in a source file and library with the RTPA generic program name *ALL will audit enable all programs executed in a job, at all levels.



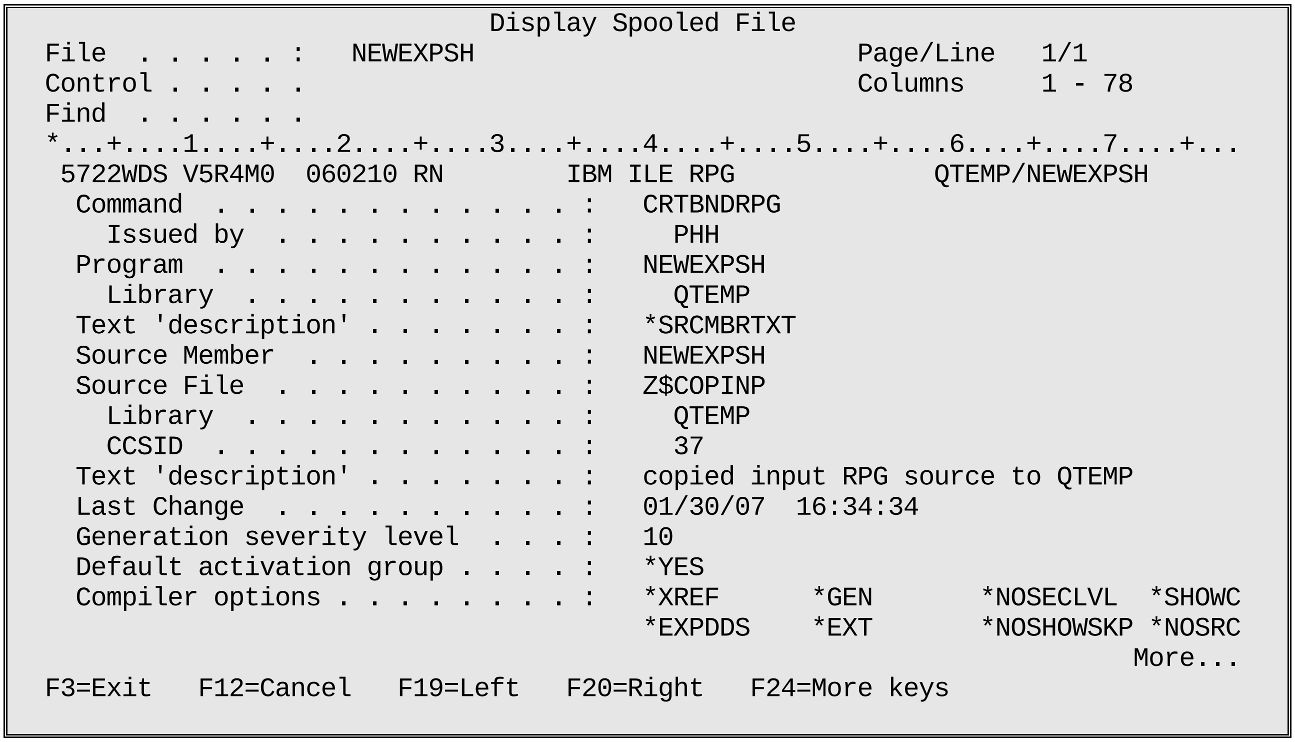
```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  6/04/07
                         Select Called Programs to Audit           Time: 16:43:06
   Program NEWEXPSH
  Type choices, press Enter.
   Y=Include in audit
                                                                  C
  Opt Seq# Called Pgm Called program description                  B Comment
  Y    534 BATCHPGM1  satch program with call to another batch pr   CALL BATCH PR
  Y    706 Z$PGM01C   RETRIEVE USER PROFILE TEXT                    GET USER PROF
  Y    926 TEST3      TEST SOURCE PROGRAM 3 RPGIV batch program W   CALL BATCH PR










  F3=Exit       F12=Cancel    Enter=Accept options and continue
                                           (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 2.4 RTPA called programs from program NEWEXPSH

Entering a 5 to the left of the Input will display the input source program compile listing.

Entering a 5 to the left of the Expand will display the expanded RTPA audit enabled compile listing, illustrating exactly how RTPA enables RPG programs for program auditing.

RTPA compiles the input RPG source program to ensure a valid RPG compile, and to obtain a compile listing, which includes all copybook and SQL expansions and a field cross reference listing.

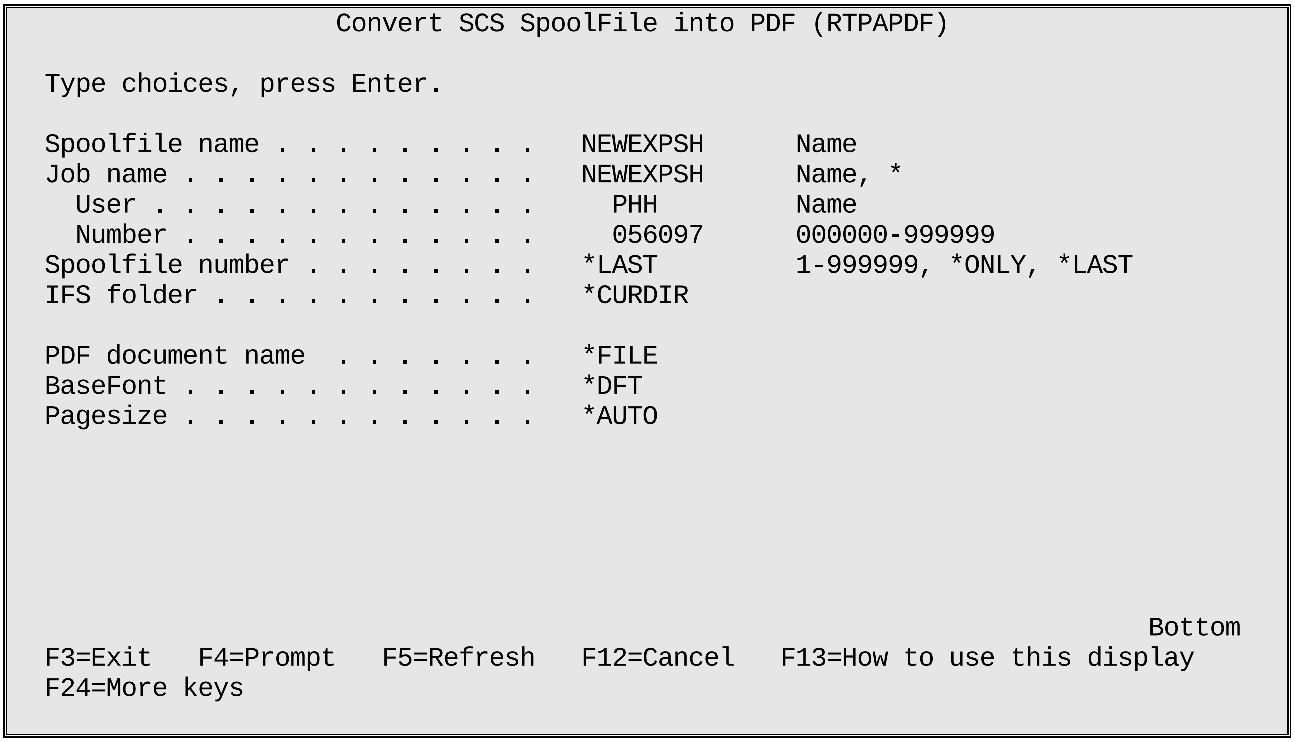


```
                            Display Spooled File
File  . . . . . :   NEWEXPSH                        Page/Line   1/1
Control . . . . .                                   Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 5722WDS V5R4M0  060210 RN        IBM ILE RPG            QTEMP/NEWEXPSH
  Command  . . . . . . . . . . . . . :   CRTBNDRPG
    Issued by  . . . . . . . . . . :     PHH
  Program  . . . . . . . . . . . . . :   NEWEXPSH
    Library  . . . . . . . . . . . :     QTEMP
  Text 'description' . . . . . . . . :   *SRCMBRTXT
  Source Member  . . . . . . . . . . :   NEWEXPSH
  Source File  . . . . . . . . . . . :   Z$COPINP
    Library  . . . . . . . . . . . :     QTEMP
    CCSID  . . . . . . . . . . . . :     37
  Text 'description' . . . . . . . . :   copied input RPG source to QTEMP
  Last Change  . . . . . . . . . . . :   01/30/07  16:34:34
  Generation severity level  . . . . :   10
  Default activation group . . . . . :   *YES
  Compiler options . . . . . . . . . :   *XREF      *GEN       *NOSECLVL  *SHOWC
                                         *EXPDDS    *EXT       *NOSHOWSKP *NOSRC
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 2.5 Input compile for NEWEXPSH input source program

Entering a P to the left of the Input will convert the WRKSPLF file to a searchable PDF on the IFS and display the input source program compile listing in searchable PFD if the required IBM software and Adobe Reader 7.0 is installed.

The RTPAPDF command may by used to concert RTPA audit spool file output (file ZZAUDITP) to searchable PDF on the IFS if the appropriate IBM software is installed.



```
                     Convert SCS SpoolFile into PDF (RTPAPDF)

 Type choices, press Enter.

 Spoolfile name . . . . . . . . .   NEWEXPSH      Name
 Job name . . . . . . . . . . . .   NEWEXPSH      Name, *
   User . . . . . . . . . . . . .     PHH         Name
   Number . . . . . . . . . . . .     056097      000000-999999
 Spoolfile number . . . . . . . .   *LAST         1-999999, *ONLY, *LAST
 IFS folder . . . . . . . . . . .   *CURDIR

 PDF document name  . . . . . . .   *FILE
 BaseFont . . . . . . . . . . . .   *DFT
 Pagesize . . . . . . . . . . . .   *AUTO






                                                                      Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 2.6 Convert RTPA input source program NEWEXPSH compile into PDF in the IFS

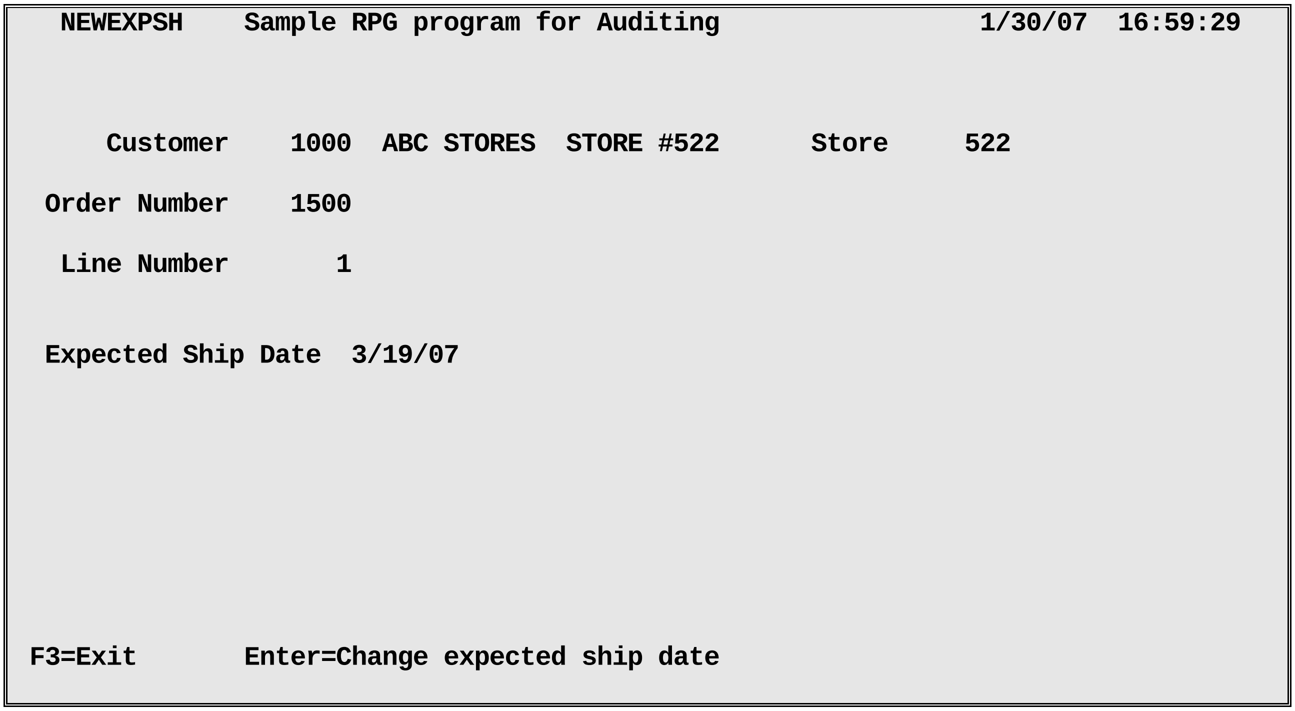
Spool file NEWEXPSH converted into /NEWEXPSH.pdf.   (in the System i IFS)

The Expanded RTPA compile listing shows the inserted RTPA Z$ audit statements from the copied input source member, which is always in a source member in library Z$AUDITE.

Execute the Program

- Execute the audit-enabled program NEWEXPSH using the expanded obkect program (in library Z$AUDITE) by typing on the command line:

```
CALL Z$TEST1N


(This is a CLP that calls program NEWEXPSH passing
order number and line number parameters)
```



```
   NEWEXPSH    Sample RPG program for Auditing                1/30/07  16:59:29



      Customer    1000  ABC STORES  STORE #522      Store     522

  Order Number    1500

   Line Number       1

  Expected Ship Date  3/19/07








 F3=Exit       Enter=Change expected ship date
```

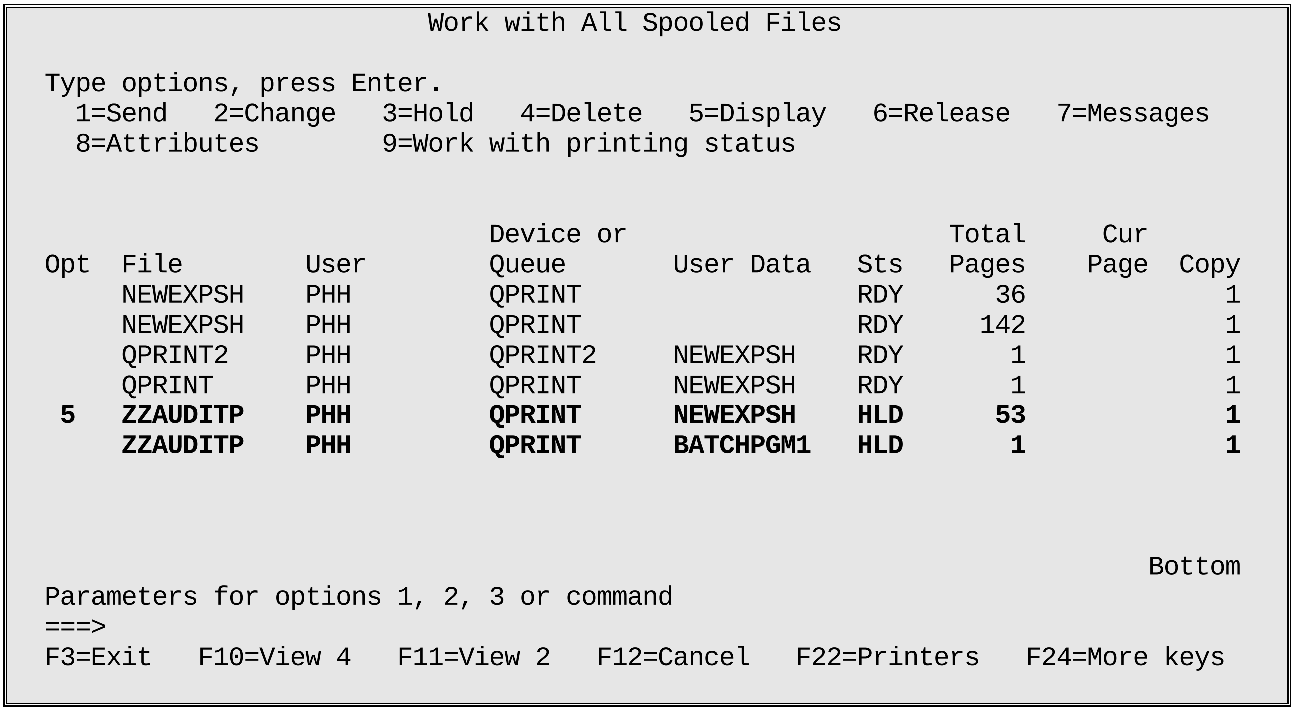
Figure 2.7 The audit enabled program NEWEXPSH executes, recording audits in real-time

Review the Audit File (RTPA audit output file ZZAUDITP)

The NEWEXPSH program object created an audit file in printer file ZZAUDITP, which you will identify by the user data NEWEXPSH in the user outq, or whatever the outq was specified in the RTPA main screen.

❖ Use the IBM WRKSPLF command to display the audit file. At the command line, type:

```
WRKSPLF
```



```
                           Work with All Spooled Files

 Type options, press Enter.
   1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
   8=Attributes        9=Work with printing status


                                   Device or                       Total     Cur
 Opt  File         User         Queue       User Data   Sts    Pages    Page  Copy
      NEWEXPSH     PHH          QPRINT                  RDY       36              1
      NEWEXPSH     PHH          QPRINT                  RDY      142              1
      QPRINT2      PHH          QPRINT2     NEWEXPSH    RDY        1              1
      QPRINT       PHH          QPRINT      NEWEXPSH    RDY        1              1
  5   ZZAUDITP     PHH          QPRINT      NEWEXPSH    HLD       53              1
      ZZAUDITP     PHH          QPRINT      BATCHPGM1   HLD        1              1



                                                                              Bottom
 Parameters for options 1, 2, 3 or command
 ===>
 F3=Exit   F10=View 4   F11=View 2   F12=Cancel   F22=Printers   F24=More keys
```

Figure 2.8 Display ZZAUDITP audit output of program NEWEXPSH

The spool file shows the compile listing for the NEWEXPSH input source program (36 pages), and for the NEWEXPSH RTPA audit enabled compile listing (142 pages) from the RTPA expand of the input source program.

The NEWEXPSH program produces two out print files (QPRINT2 and QPRINT).

RTPA for RPG produces a ZZAUDITP audit report of 53 pages showing every executing source statement, the data processed, and the exact time the statement was executed.

RTPA for RPG also audits all called programs from the expanded source program, if the called programs have been also expanded by RTPA. Program BATCHPGM1 is audited (because it was also previously expanded by RTPA), and is called from program NEWEXPSH on page 22 of the audit output.

- ❖ Review the NEWEXPSH 53 page entry of the spool file listing and select **option 5** to display the RTPA audit output. (The RTPAPDF command can by used to create a searchable PDF of this output on the IFS)
- ❖ See Appendix E of this manual for a complete audit of the expanded program NEWEXPSH.
- ❖ The ZZAUDITP audit print file is 198 characters.

(From the NEWEXPSH audit output in Appendix E)

```
Program: NEWEXPSH  New Expected Ship Date RPGIV   Obj Lib: Z$AUDITE    Initiated:  12/09/06 11.10.48.733   PAGE   1
         NEWEXPSH    NEWEXPSH
Job: 026982              User Profile: PHH                             Source File/Library:  QRPGLESRC Z$AUDIT
Line#                                                                              Do# SrcId ChgDat   Seq#   Time
 956 C     *INZSR        BEGSR                                                         ph456 011227  89700 11.10.48.742
 957  * initialize fields and arrays                                                         060318  89800 11.10.48.742
 958 C                   MOVEL     *BLANKS       MOVSW            1                    ph456 011227  89900 11.10.48.742
 959 C                   Z-ADD     12            $$D                                   ph456 011227  90000 11.10.48.751
                                                 121212121212121212121212
 960 C                   MOVEA     *ZERO         $$D2                                  ph456 011227  90100 11.10.48.751
```

```
                                                 000000000000000000000000000000000000000
 961 C                   MOVEA     1111          $$D3                              ph457 011227  90200 11.10.48.751
                                                 1111000000000000000000000000000000000000000000000000
 962 C                   MOVEL     '88888888'    $$A                               ph456 011227  90300 11.10.48.751
 963 C                   ENDSR                                                     ph456 011227  90400 11.10.48.751
 229  * CUSTOMER MASTER KEY                                                              991225  17600 11.10.48.751
 234  * RECEIVE PARAMETERS PASSED FROM CALLING PROGRAM                                   990918  18100 11.10.48.751
 235 C     *ENTRY        PLIST                                                           990918  18200 11.10.48.751
 236 C                   PARM                    PARMIN           44                     991225  18300 11.10.48.751
                                                 000150000001
 238  * INPUT DATA AREA TSTDTA                                                           010126  18500 11.10.48.751
 240 C                   IN        TSTDTA                                                010126  18700 11.10.48.751
VAR TSTDTA                  1 -    100  ABCDEFGHIJKLMNOPQRSTUVWZYZ012345                      BBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
C
VAR TSTDTA               101 -    200   CCCCCCCCCCCCCCCCCCCCCCCCCCCCC                         DDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
E
VAR TSTDTA            201 -   256  EEEEEEEEEEEEEEEEEEEEEEEEEEEEE                   FFFFFF
 242 C                   TIME                    TIMES             6 0                   000302  18900 11.10.48.751
                                                111048
 244  * MOVE INPUT PARM INTO FOUR PARM FIELDS IN DS                                      990918  19100 11.10.48.751
 245 C                   MOVEL     PARMIN        PARMRE                                  990918  19200 11.10.48.752
                                   000150000001
                                                 000150000001
 247  * start free form                                                                  061201  19301 11.10.48.752
 249   // now in free form RPG                                                           061201  19401 11.10.48.752
 250     torder = 1500;                                                                  020623  19500 11.10.48.752
           1500
 251       iorder = 78.543;                                                              020623  19600 11.10.48.752
           78.543
 252    // value of iorder has now been computed                                         061201  19601 11.10.48.752
 253         xorder = torder + 13.45  +                                                  020623  19700 11.10.48.753
            1618.19     1500
 254     // this is a continuation free form statement preceeded with +                  061201  19701 11.10.48.753
 255                   26.2 + iorder;                                                    020623  19800 11.10.48.753
                              78.543
 256   // end of free form                                                               061201  19801 11.10.48.753
 258  * resume fixed format calc statements                                              061201  19901 11.10.48.753
 259 c                   eval      rorder = iorder +98                                   020623  20000 11.10.48.753
                                  176.543
                                            78.543
 260  * add 30 days to start_date to get end_date                                        020623  20100 11.10.48.753
 261 c     start_date    adddur    30:*days      end_date                                020623  20200 11.10.48.753
           1998-12-18                            1999-01-17
 262  * add 1 month to end_date                                                          020623  20300 11.10.48.753
 263 c                   adddur    1:*months     end_date                                020623  20400 11.10.48.753
                                                 1999-02-17
 264  * extract day number from date                                                     050516  20500 11.10.48.754
 265 c                   extrct    end_date:*D   dayno             2 0          dd       050516  20600 11.10.48.754
                                   1999-02-17
                                                    17
 266  * extract month number from date                                                   050517  20700 11.10.48.754
 267 c                   extrct    end_date:*M   Month_no          2 0          mm       050517  20800 11.10.48.754
                                   1999-02-17
                                                        2
 268  * extract year number from date                                                    050517  20900 11.10.48.754
 269 c                   extrct    end_date:*Y   Year_no           4 0          ccyy     050517  21000 11.10.48.754
                                   1999-02-17
                                                    1999
 270  * add 1 year to start_date                                                         020623  21100 11.10.48.754
 271 c     employ_dat    adddur    1:*years      anniv_dat                               020623  21200 11.10.48.754
           12/29/1992                            12/29/1993
 272  * add 3 hours, 22 minutes and 50 seconds to midnight                               020623  21300 11.10.48.754
 273 c     T'00.00.00'   adddur    3:*hours      end_time                                020623  21400 11.10.48.754
                                                 03:00:00
 274 c                   adddur    22:*minutes   end_time                                020623  21500 11.10.48.754
                                                 03:22:00
 275 c                   adddur    50:*seconds   end_time                                020623  21600 11.10.48.754
                                                 03:22:50
 276  * add 1000 microseconds to a time stamp (26 character date and time)               020623  21700 11.10.48.754
 277 c                   adddur    1000:*ms      total_time                              020623  21800 11.10.48.758
                                                 0001-01-01-00.00.00.001000
                                                                                                       11.10.48.733   PAGE
 278 C                   Z-ADD     14.25         TESD             12 3                   010113  21900 11.10.48.758
                                               14.250
 279 C                   Z-ADD     *zero         LocatTotal                              050415  22000 11.10.48.758

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 281  * MOVE INPUT ORDER AND LINE TO KORDER AND LINE (KEYED)                             011227  22200 11.10.48.760
 282 C                   Z-ADD     PORDER        KORDER                                  020623  22300 11.10.48.760
                                     1500
                                                   1500
 283 C                   Z-ADD     PLINE         KLINE                                   001007  22400 11.10.48.761
                                       1
                                                     1
 284  *----------------------------------------------------------------                  000323  22500 11.10.48.761
 285  * DISPLAY HEADING SCREEN                                                           000323  22600 11.10.48.761
 286  *----------------------------------------------------------------                  000323  22700 11.10.48.761
 287 C     DISP01        TAG                                                             000514  22800 11.10.48.761
 288  * CLEAR EXPECTED SHIP DATE AND ERROR CODE                                          000514  22900 11.10.48.761
 289 C                   Z-ADD     *ZERO         PEXPSH                                  000514  23000 11.10.48.761
                                                      0
 290 C                   MOVEL     *BLANKS       PERROR                                  000514  23100 11.10.48.761
 291 C                   Z-ADD     *ZEROS        KCUSNO                                  001002  23200 11.10.48.761
                                                      0
 292 C                   Z-ADD     *ZEROS        KSTORE                                  001002  23300 11.10.48.761
                                                      0
 293 C                   MOVEL     *BLANKS       KCUSNA                                  000323  23400 11.10.48.761
 294 C                   Z-ADD     *ZERO         EXPMDY                                  000323  23500 11.10.48.761
                                                      0
 295 C                   TIME                    TIMEN             6 0                   010501  23600 11.10.48.761
                                                111048
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.10.48.761 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-120906 TIMEN-111048
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.10.54.917 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111048
 297  * TEST F3                                                                          000323  23800 11.10.54.917
 298 C     *IN03         CABEQ     *ON           DONE                                    000323  23900 11.10.54.917
           0
 300 C     UDATE         CABEQ     090100        DONE                                    010113  24100 11.10.54.917
          120906
```

Figure 2.9 Audit Output File for NEWEXPSH

The RTPA audit output for RPGLE shows the exact time the statement was executed to the millisecond.

The EXFMT (execute Format) instruction is a WRITE to the screen, then a READ from the screen (with keyed data or command keys). RTPA audits both the WRITE and the READ, with the exact times of each, and shows the contents of the screen variables and variables.

Thus RTPA for RPG auditing shows and records the time the user took to enter the data and to press the enter key, or a command key, and RTPA records exactly what was keyed.
The user changed the order line number (KLINE) from 1 to 2, and took a little over 3 seconds to press the enter key.

Note that RTPA auditing shows the contents of all variables processed, the status of all command keys used, and the contents of KLISTS and Parameters.

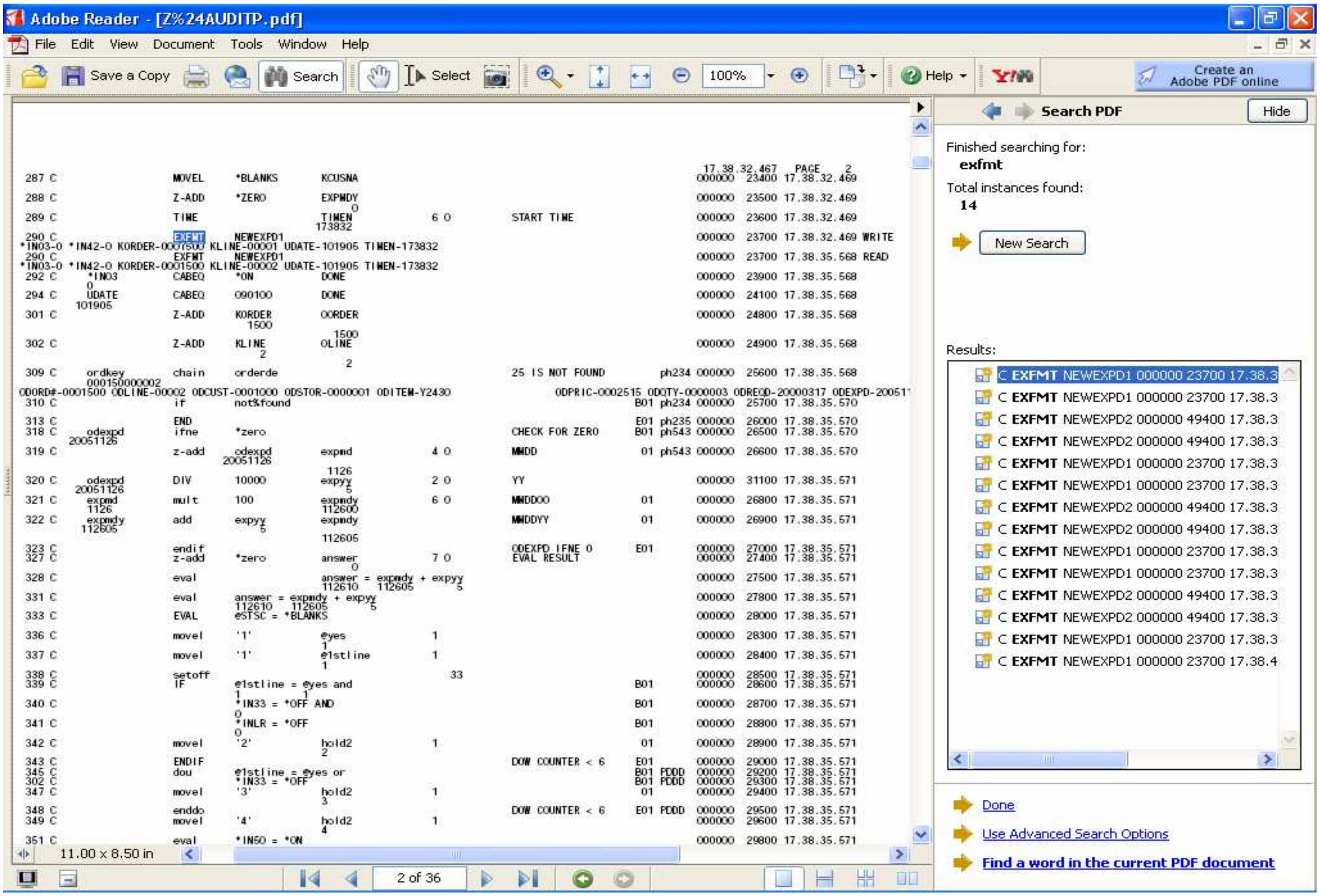
```
 296 C                   EXFMT     NEWEXPD1                                     051007  23700 11.10.48.761 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-120906 TIMEN-111048
 296 C                   EXFMT     NEWEXPD1                                     051007  23700 11.10.54.917 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111048
 297  * TEST F3                                                                 000323  23800 11.10.54.917
 298 C     *IN03         CABEQ     *ON           DONE                           000323  23900 11.10.54.917
           0
```

Figure 2.10 Audit Output of NEWEXPSH showing the data keyed and the elapsed time to enter it

RTPA auditing from the input source statement (and the input program compile listing) provides powerful capabilities to customize the audit output to achieve desired auditing analysis results, as is illustrated but the **double audit of the EXFMT (Execute Format) operation code**. Thus the programmer or auditor can see the display file record variable contents, command keys and exact time the screen was displayed (Write), and then the display file record variable contents, command keys and exact time when the screen was read (READ).

Program NEWEXPSH audit output in searchable PDF

The user can search an RTPA audit output PDF for all the EXFMT operations actually executed, as in the following figure. The user may search the RTPA audit output on any string of characters, including data values.



Adobe Reader - [Z%24AUDITP.pdf]
File Edit View Document Tools Window Help
Save a Copy
Search
Select
100%
Help
Create an Adobe PDF online
Search PDF
Hide
11.00 x 8.50 in
2 of 36

```
                                                                                 17.38.32.467   PAGE    2
 287 C                MOVEL     *BLANKS      KCUSNA                              000000  23400 17.38.32.469
 288 C                Z-ADD     *ZERO        EXPMDY                              000000  23500 17.38.32.469
                                                  0
 289 C                TIME                   TIMEN           6 0    START TIME        000000  23600 17.38.32.469
                                            173832
 290 C                EXFMT     NEWEXPD1                                         000000  23700 17.38.32.469 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-101905 TIMEN-173832
 290 C                EXFMT     NEWEXPD1                                         000000  23700 17.38.35.568 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-101905 TIMEN-173832
 292 C     *IN03      CABEQ     *ON          DONE                                000000  23900 17.38.35.568
           0
 294 C     UDATE      CABEQ     090100       DONE                                000000  24100 17.38.35.568
         101905
 301 C                Z-ADD     KORDER       OORDER                              000000  24800 17.38.35.568
                                  1500
                                               1500
 302 C                Z-ADD     KLINE        OLINE                               000000  24900 17.38.35.568
                                    2
                                                2
 309 C     ordkey     chain     orderde                 25 IS NOT FOUND     ph234 000000  25600 17.38.35.568
           000150000002
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430        ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317 ODEXPD-20051
 310 C                if        not%found                               B01 ph234 000000  25700 17.38.35.570

 313 C                END                                               E01 ph235 000000  26000 17.38.35.570
 318 C     odexpd     ifne      *zero               CHECK FOR ZERO      B01 ph543 000000  26500 17.38.35.570
         20051126
 319 C                z-add     odexpd       expmd           4 0    MMDD            01 ph543 000000  26600 17.38.35.570
                               20051126
                                              1126
 320 C     odexpd     DIV       10000        expyy           2 0    YY                    000000  31100 17.38.35.571
          20051126                               5
 321 C     expmd      mult      100          expmdy          6 0    MMDDOO          01        000000  26800 17.38.35.571
           1126                              112600
 322 C     expmdy     add       expyy        expmdy                 MMDDYY          01        000000  26900 17.38.35.571
          112605                    5
                                             112605
 323 C                endif                                     ODEXPD IFNE 0   E01       000000  27000 17.38.35.571
 327 C                z-add     *zero        answer          7 0    EVAL RESULT               000000  27400 17.38.35.571
                                                  0
 328 C                eval                   answer = expmdy + expyy                  000000  27500 17.38.35.571
                                             112610   112605     5
 331 C                eval      answer = expmdy + expyy                               000000  27800 17.38.35.571
                                112610   112605     5
 333 C                EVAL      @STSC = *BLANKS                                      000000  28000 17.38.35.571

 336 C                movel     '1'          @yes            1                        000000  28300 17.38.35.571
                                             1
 337 C                movel     '1'          @1stline        1                        000000  28400 17.38.35.571
                                             1
 338 C                setoff                                   33                     000000  28500 17.38.35.571
 339 C                IF        @1stline = @yes and                     B01           000000  28600 17.38.35.571
                                1          1
 340 C                          *IN33 = *OFF AND                        B01           000000  28700 17.38.35.571
                                0
 341 C                          *INLR = *OFF                            B01           000000  28800 17.38.35.571
                                0
 342 C                movel     '2'          hold2           1                  01    000000  28900 17.38.35.571
                                             2
 343 C                ENDIF                              DOW COUNTER < 6    E01           000000  29000 17.38.35.571
 345 C                dou       @1stline = @yes or                      B01 PDDD      000000  29200 17.38.35.571
 302 C                          *IN33 = *OFF                            B01 PDDD      000000  29300 17.38.35.571
 347 C                movel     '3'          hold2           1                  01    000000  29400 17.38.35.571
                                             3
 348 C                enddo                              DOW COUNTER < 6    E01 PDDD      000000  29500 17.38.35.571
 349 C                movel     '4'          hold2           1                        000000  29600 17.38.35.571
                                             4
 351 C                eval      *IN50 = *ON                                          000000  29800 17.38.35.571
```

Finished searching for:
exfmt

Total instances found:
14

New Search

Results:

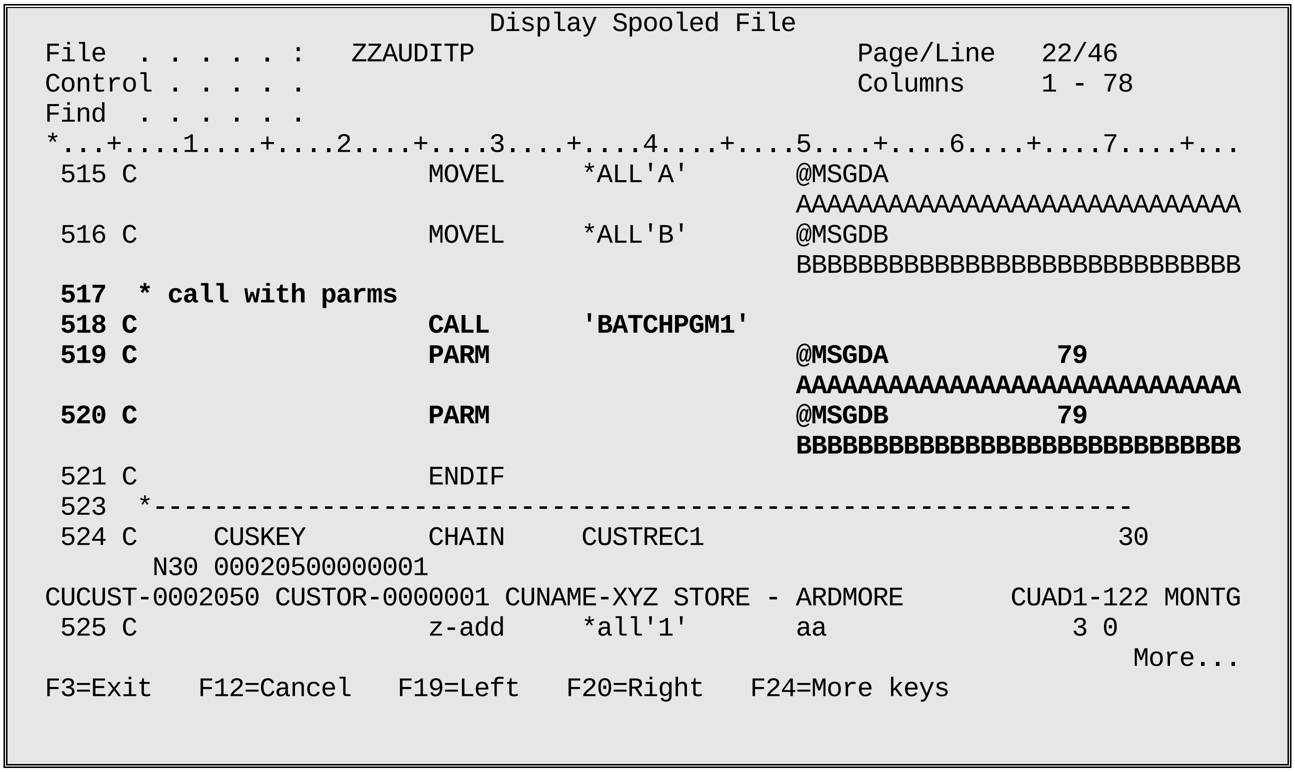
```
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.4
```

Done

Use Advanced Search Options

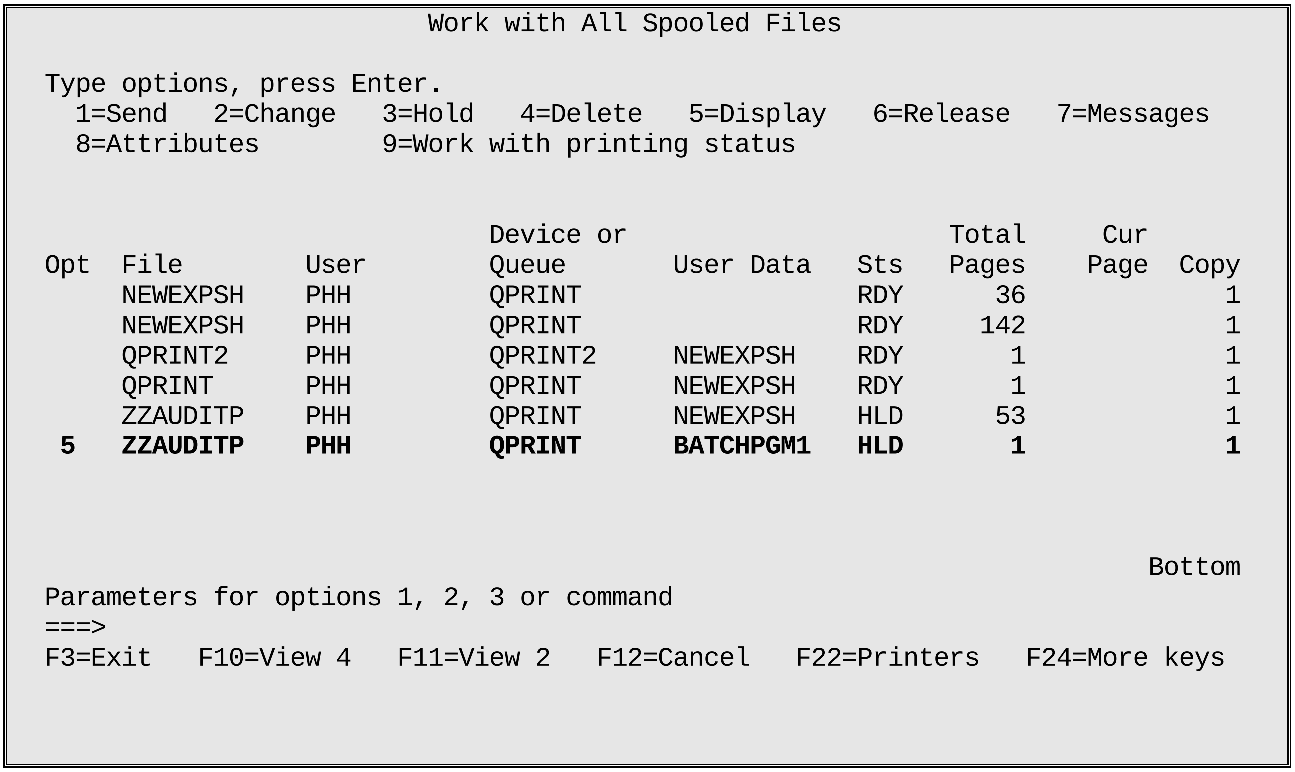
Find a word in the current PDF document

Figure 2.11 Audit Output of NEWEXPSH showing the data keyed and the elapsed time to enter it in PDF



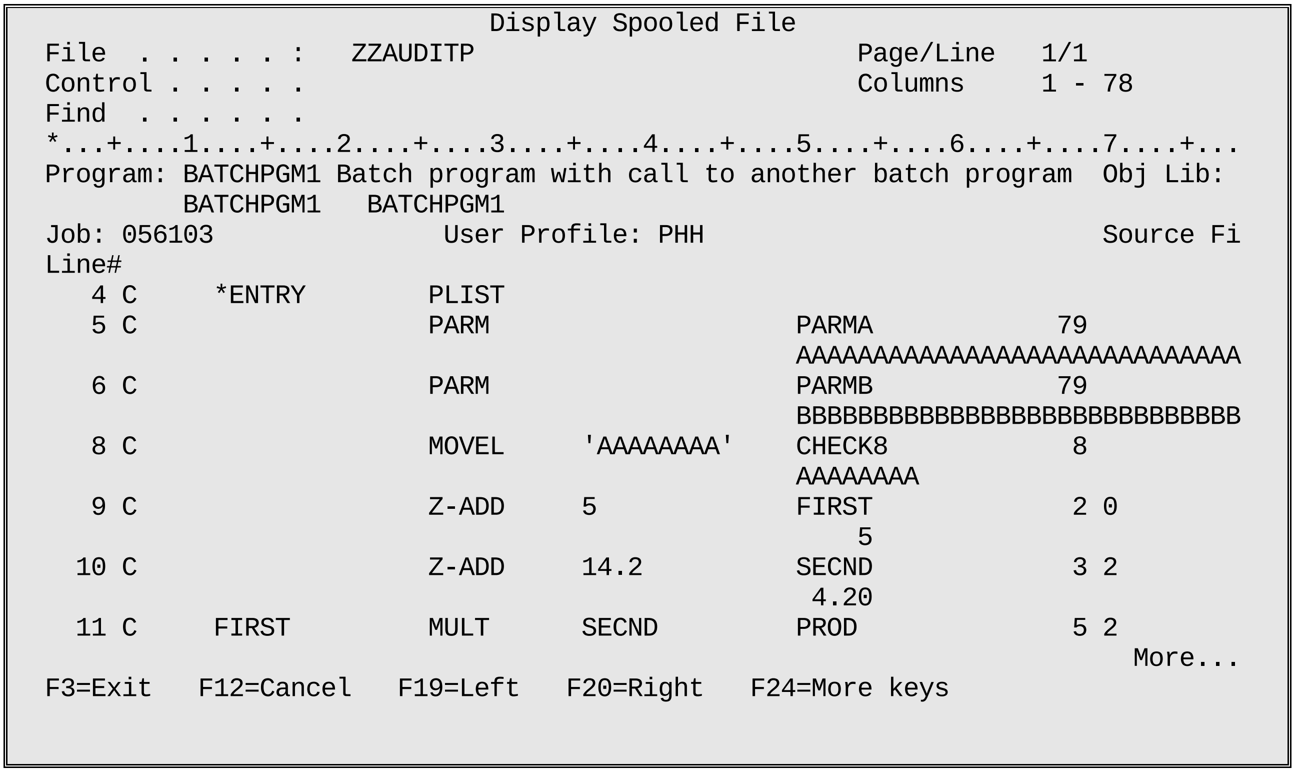
```
                               Display Spooled File
File  . . . . . :   ZZAUDITP                         Page/Line   22/46
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 515 C                   MOVEL     *ALL'A'       @MSGDA
                                                 AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 516 C                   MOVEL     *ALL'B'       @MSGDB
                                                 BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
 517  * call with parms
 518 C                   CALL      'BATCHPGM1'
 519 C                   PARM                    @MSGDA           79
                                                 AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 520 C                   PARM                    @MSGDB           79
                                                 BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
 521 C                   ENDIF
 523  *------------------------------------------------------------------
 524 C     CUSKEY        CHAIN     CUSTREC1                          30
      N30 00020500000001
CUCUST-0002050 CUSTOR-0000001 CUNAME-XYZ STORE - ARDMORE       CUAD1-122 MONTG
 525 C                   z-add     *all'1'       aa                3 0
                                                                      More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 2.12 Audit of CALL to program BATCHPGM1 in program NEWEXPSH



```
                          Work with All Spooled Files

Type options, press Enter.
  1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
  8=Attributes        9=Work with printing status


                            Device or                         Total     Cur
Opt  File        User        Queue       User Data   Sts    Pages    Page  Copy
     NEWEXPSH    PHH         QPRINT                  RDY       36             1
     NEWEXPSH    PHH         QPRINT                  RDY      142             1
     QPRINT2     PHH         QPRINT2     NEWEXPSH    RDY        1             1
     QPRINT      PHH         QPRINT      NEWEXPSH    RDY        1             1
     ZZAUDITP    PHH         QPRINT      NEWEXPSH    HLD       53             1
 5   ZZAUDITP    PHH         QPRINT      BATCHPGM1   HLD        1             1



                                                                        Bottom
Parameters for options 1, 2, 3 or command
===>
F3=Exit   F10=View 4   F11=View 2   F12=Cancel   F22=Printers   F24=More keys
```

Figure 2.13 Display ZZAUDITP audit output of program BATCHPGM1



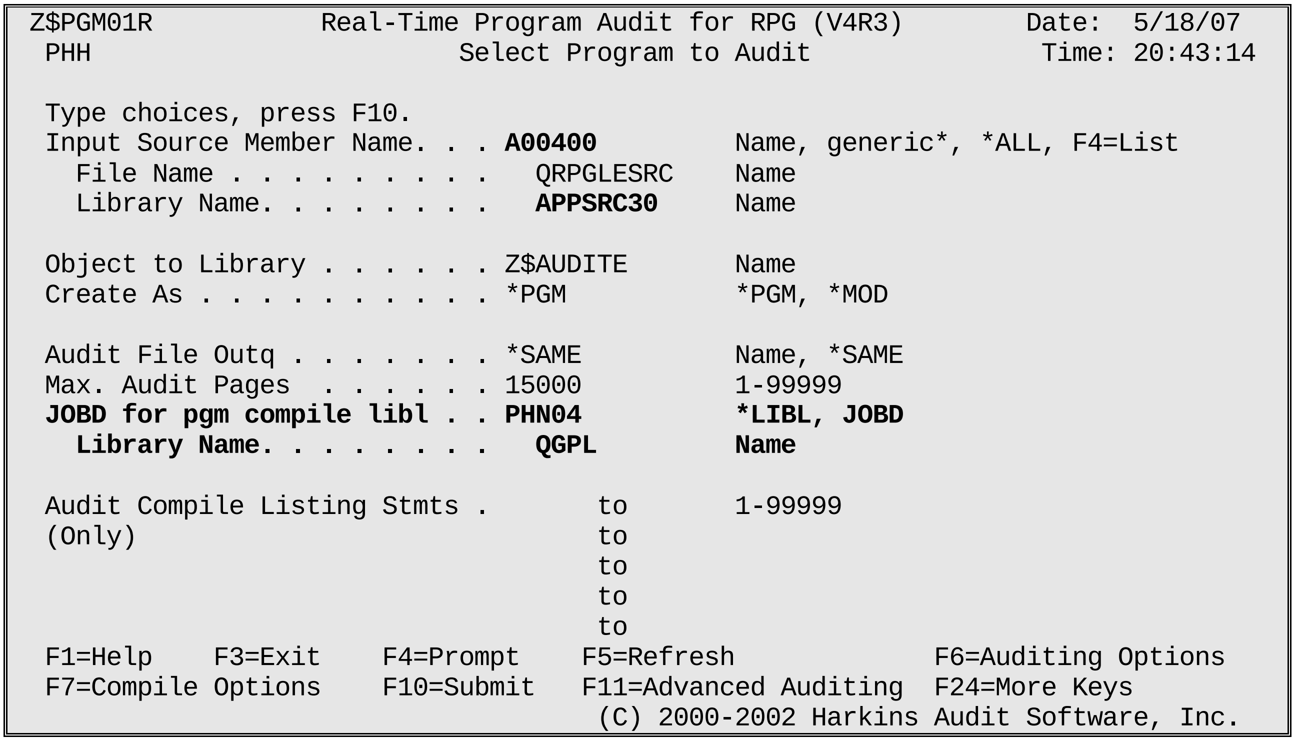
```
                               Display Spooled File
File  . . . . . :   ZZAUDITP                        Page/Line   1/1
Control . . . . .                                   Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program: BATCHPGM1 Batch program with call to another batch program  Obj Lib:
         BATCHPGM1   BATCHPGM1
Job: 056103              User Profile: PHH                          Source Fi
Line#
   4 C     *ENTRY        PLIST
   5 C                   PARM                    PARMA            79
                                                 AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
   6 C                   PARM                    PARMB            79
                                                 BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
   8 C                   MOVEL     'AAAAAAAA'    CHECK8            8
                                                 AAAAAAAA
   9 C                   Z-ADD     5             FIRST             2 0
                                                     5
  10 C                   Z-ADD     14.2          SECND             3 2
                                                  4.20
  11 C     FIRST         MULT      SECND         PROD              5 2
                                                                       More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 2.14 ZZAUDITP audit output of program BATCHPGM1 called from program NEWEXPSH

Program BATCHPGM1 audit output

```
Program: BATCHPGM1 Batch program with call to     Obj Lib: Z$AUDITE    Initiated:  01/30/07 17.05.42.106   PAGE   1
         BATCHPGM1   BATCHPGM1
Job: 056103               User Profile: PHH                          Source File/Library:  QRPGLESRC Z$AUDIT
Line#                                                                    Do# SrcId  ChgDat  Seq#   Time
   4 C     *ENTRY        PLIST                                                      010529   400
   5 C                   PARM                    PARMA           79                 010529   500 17.05.42.126
                                                 AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
   6 C                   PARM                    PARMB           79                 010529   600 17.05.42.126
                                                 BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
   8 C                   MOVEL     'AAAAAAAA'    CHECK8           8                 010827   800 17.05.42.131
                                                 AAAAAAAA
   9 C                   Z-ADD     5             FIRST            2 0               000521    900 17.05.42.131
                                                     5
  10 C                   Z-ADD     14.2          SECND            3 2               000521   1000 17.05.42.131
                                                  4.20
  11 C     FIRST         MULT      SECND         PROD             5 2               000521   1100 17.05.42.131
              5                    4.20
                                                21.00
------------------------------------------------------------------------------------------------------------
```

Figure 2.15 ZZAUDITP audit output of program BATCHPGM1 (Appendix E)



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  5/18/07
 PHH                         Select Program to Audit               Time: 20:43:14

 Type choices, press F10.
 Input Source Member Name. . . A00400         Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QRPGLESRC    Name
   Library Name. . . . . . . .   APPSRC30     Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 15000          1-99999
 JOBD for pgm compile libl . . PHN04          *LIBL, JOBD
   Library Name. . . . . . . .   QGPL         Name

 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F1=Help    F3=Exit    F4=Prompt    F5=Refresh             F6=Auditing Options
 F7=Compile Options    F10=Submit   F11=Advanced Auditing  F24=More Keys
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 2.16 Expand Program A00400 from library APPSRC30 with JOBD PHN04 in library QGPL

Chapter

3

Chapter 3: **RTPA Overview – Auditing Concepts**

This chapter covers the concepts, commands and basic conventions to start auditing software with RTPA.

RTPA is intuitive and easy to use. This chapter explains general concepts that you should understand to maximize the value you get from using RTPA.

For step-by-step instructions on using the RTPA features, turn to:
Chapter 4: Using RTPA

Auditable Information

RTPA can capture just about everything that your program does when it executes. RTPA also allows you to refine your audit to include only the information that you want.

Auditable information includes:

- Source statements
- Comments
- Variable contents
- Compile listing statement sequence number
- Change ID of the source statement (positions one through five of the source statement)
- Statement change date
- Time of execution

Creating an Audit – Overview

RTPA is a software utility that programmers use to create audits of their programs. It may be useful to think of RTPA as a *pre-compiler* because the bulk of its work is done prior to compiling the object. RTPA's process is remarkably simple:

1. RTPA analyses the source code and creates a new, temporary source code file (called the **audit-enabled source code**) containing both the source code and **audit statements**.

2. RTPA compiles the audit-enabled source code with the regular compiler. The resulting object is an **audit-enabled executable object program**.

3. When the audit-enabled executable object program is initiated (interactive or batch), the executable itself produces an **audit output file**, which we normally refer to as an **audit file** or **audit**.

Audit-Enabling A Program

Using the RTPA interface, a programmer selects source files for auditing, chooses what types of auditable information to include in the audit file and the conditions under which the information should be audited. (The default is to include all information under any condition.)

Once the software and options are selected, RTPA temporarily creates an audit-enabled source file in QTEMP.

RTPA then compiles the audit-enabled source program using the OS/400 compiler, putting the audit-enabled executable object into the library that you selected on the main screen.

Producing an Audit File

When the audit-enabled executable object is executed, it automatically produces audit output, which is sent to the printer file ZZAUDITP.

Reviewing an Audit File

The audit file is sent to the printer file ZZAUDITP. You can print out that file or use WRKSPLF to display the spooled file.

Audit Statement Ordering

By default, RTPA audits some statements prior to execution and audits other statements after execution.

Data Modifying Statements

By default, RTPA audits data modifying statements (i.e., ADD, MULTIPLY, MOVE, CHAIN) after they are executed. The audit produced this way reflects the data results of the statement.

EVAL Statements

The EVAL, DOW and WHEN statements are audited after the EVAL and *all* continuation statements (i.e., AND/OR) for the EVAL are executed.

Branching and Conditional Statements

Branching and Conditional RPG operations (i.e., EXSR, GOTO, IF, RETURN) are audited before the source statement is executed.

Special Case – Uninitialized Fields

Numeric fields defined as define storage (DS) fields (data type Z or zoned decimal in RPGIII and data type S in RPGIV) that are located in an IF statement are not audited prior to execution. This is done to avoid possible decimal data errors. Zoned decimal (data types Z and S) fields are audited in all other source program statements.

The DO UNTIL DOUXX (RPGIII) and DOU (RPGIV) operation statement code does not show the content of the variables.

Note – Compiler override statements in the RPGLE program may also be used to ignore decimal data errors. Also RTPA compile override options may be used to bypass decimal data errors.

Chapter

4

Chapter 4: **Using RTPA**

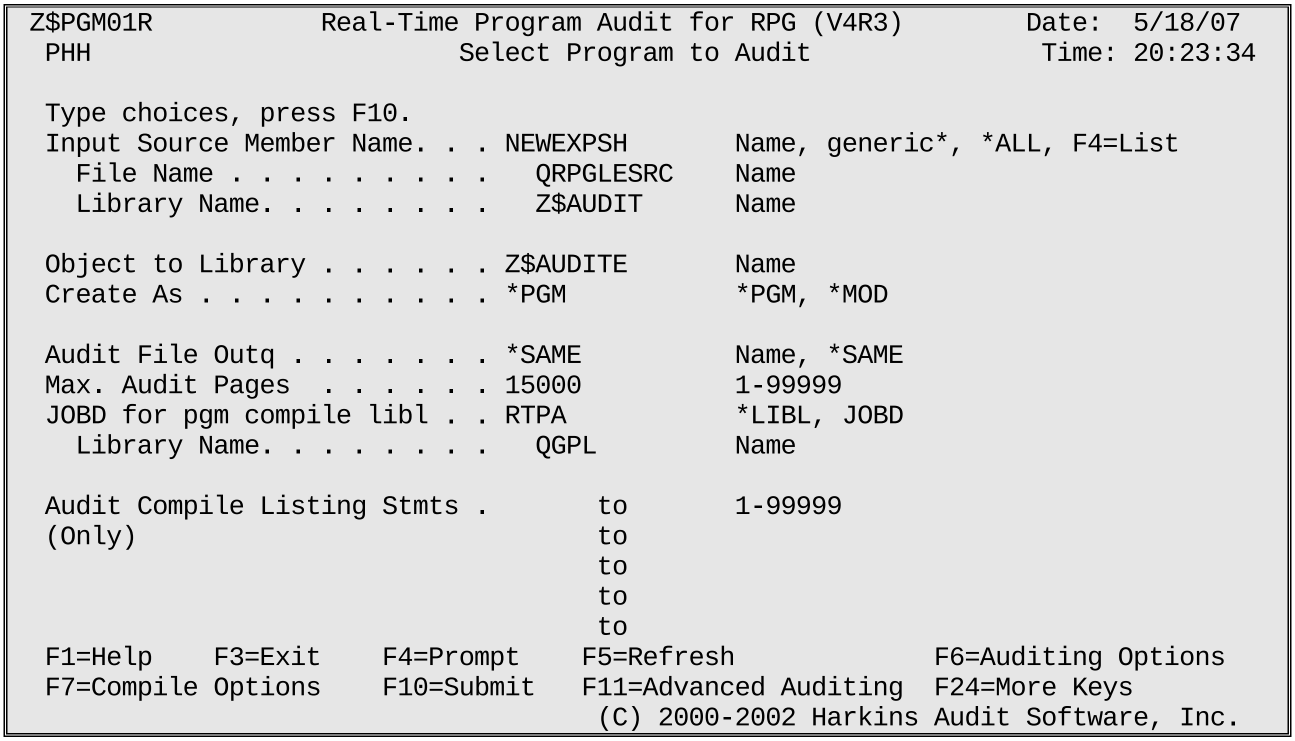
This chapter explains RTPA, a utility program that allows you to perform all of the functions required to create audit-enabled programs.

You can use RTPA to audit-enable specific programs after you have finished editing them with PDM and are sure that they compile correctly.

❖ To get the RTPA main screen, at the command line, type:

```
RTPA
```

The following screen will be displayed, as shown in Figure 4.1:



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  5/18/07
 PHH                        Select Program to Audit               Time: 20:23:34

 Type choices, press F10.
 Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QRPGLESRC    Name
   Library Name. . . . . . . .   Z$AUDIT      Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 15000          1-99999
 JOBD for pgm compile libl . . RTPA           *LIBL, JOBD
   Library Name. . . . . . . .   QGPL         Name

 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F1=Help    F3=Exit    F4=Prompt    F5=Refresh             F6=Auditing Options
 F7=Compile Options    F10=Submit   F11=Advanced Auditing  F24=More Keys
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.1 RTPA Main Selection Screen

Selecting a source member to expand for Auditing

Note – RTPA for RPG redisplays the source member name, source file, and source library from the last RTPA expansion by the user.

The first step in creating an audit is to select a source file to audit-enable (expand). The source file's type and location are dependent on the following values:

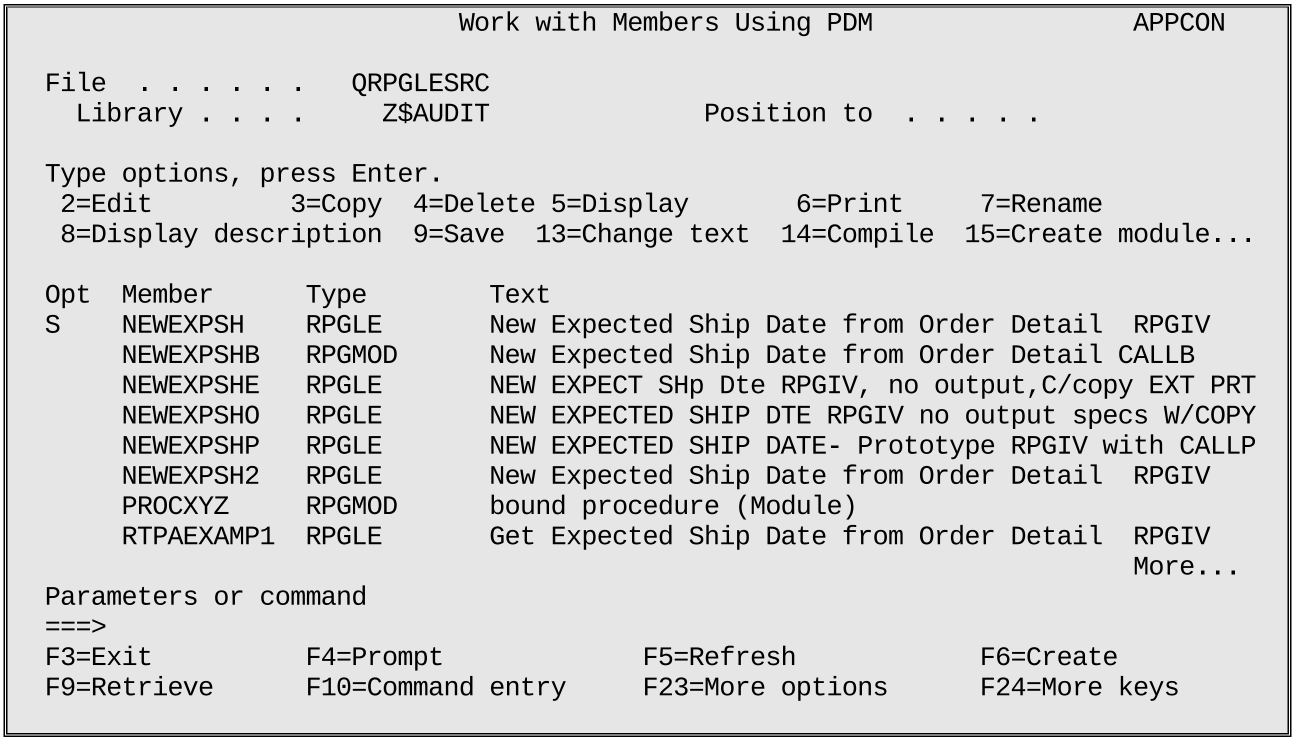
Parameter	Description
`Member Name`	The name of the member to audit
`File Name`	The source file that contains the member.
`Library Name`	The library that contains the source file.

There are two ways to select a source file:

- Type the Member Name, File Name and Library Name, or

- Select the program in PDM by blanking the program name in the RTPA main screen, then **pressing F4** to display the source members in PDM. Choose the desired source member name to return to the RTPA main screen program by **entering `S`** (S blank or blank S) next to the member name in the `Opt` column and then pressing **Enter**, then press command 3 to exit PDM, as shown in Figure 4.2. The selected PDM member name will be returned to the RTPA main screen in the source member name field.

- PDM may be used to edit or browse a source member in the RTPA main screen by pressing F4 with the cursor on the member name field.



```
                             Work with Members Using PDM                 APPCON

 File  . . . . . .   QRPGLESRC
   Library . . . .     Z$AUDIT              Position to  . . . . .

 Type options, press Enter.
  2=Edit         3=Copy  4=Delete 5=Display       6=Print     7=Rename
  8=Display description  9=Save  13=Change text  14=Compile  15=Create module...

 Opt  Member      Type        Text
 S    NEWEXPSH    RPGLE       New Expected Ship Date from Order Detail  RPGIV
      NEWEXPSHB   RPGMOD      New Expected Ship Date from Order Detail CALLB
      NEWEXPSHE   RPGLE       NEW EXPECT SHp Dte RPGIV, no output,C/copy EXT PRT
      NEWEXPSHO   RPGLE       NEW EXPECTED SHIP DTE RPGIV no output specs W/COPY
      NEWEXPSHP   RPGLE       NEW EXPECTED SHIP DATE- Prototype RPGIV with CALLP
      NEWEXPSH2   RPGLE       New Expected Ship Date from Order Detail  RPGIV
      PROCXYZ     RPGMOD      bound procedure (Module)
      RTPAEXAMP1  RPGLE       Get Expected Ship Date from Order Detail  RPGIV
                                                                        More...
 Parameters or command
 ===>
 F3=Exit          F4=Prompt             F5=Refresh            F6=Create
 F9=Retrieve      F10=Command entry     F23=More options      F24=More keys
```

Figure 4.2 Work with Members Using PDM Screen

TIP: Make sure that your program compiles correctly prior to selecting it for expansion. RTPA uses the AS/400's native compiler and may not be able to expand or compile programs with errors in them.

Selecting the Object Library for the expanded object

The expanded audit-enabled object program will be created in a library that you specify in the Object To Library field. This is the object program that will produce audit output on printer file ZZAUDITP.

Selecting the Job Description to be used for RPG source compiles

The user may enter a Job Description name and library to be used to provide the correct library list for compiles of the RPG input source program and fro the compile of the expanded RPG source programs with inserted Z$ audit statements.

```
JOBD for pgm compile libl . . RTPA           *LIBL, JOBD
   Library Name. . . . . . . .   QGPL         Name

JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
   Library Name. . . . . . . .               Name

JOBD *LIBL is to used the signed on job description library list with any added
or deleted libraries from an EDTLIBL command. This is the *CURRENT library list
```

Customizing the Audit

RTPA offers many options for determining what information is included in the audit file. While auditing everything is appropriate for many tasks, there are situations where it becomes useful to limit what is audited. You can select to include/exclude certain operations, turn the audit on or off based on specific conditions in the program, view the values of specific variables, only audit statements with a specific Change ID or Change Date, etc.

One key reason for reducing the number of audited statements is the source file size limitation of the RPGIII compiler. The RPGIII compiler has a limitation on the number of lines that a source file can contain. Because RTPA works by temporarily adding statements to the source member before compiling, it is possible that a very large source file will grow past the file size limit of the RPGIII compiler when you audit-enable it. In that case, you must refine your audit to limit the number of statements that RTPA adds to your program. (See **Chapter 8: Auditing Very Large RPG Programs** for more information on handing audits of very large files.)

Selecting Ranges of Statements to Audit

You can choose to audit up to five ranges of statements based on the program compile listing sequence numbers.

❖ On the main screen, tab to `Audit Compile Listing Statements.` Enter the input RTPA compile listing statement line whole number on which to start the audit and the statement line number on which to end the audit.

You can obtain the sequence numbers by first using RTPA to compile the original source program and reviewing the spool file compile listing. RTPA always uses the compile listing whole number line numbers, as this includes generated source statements.

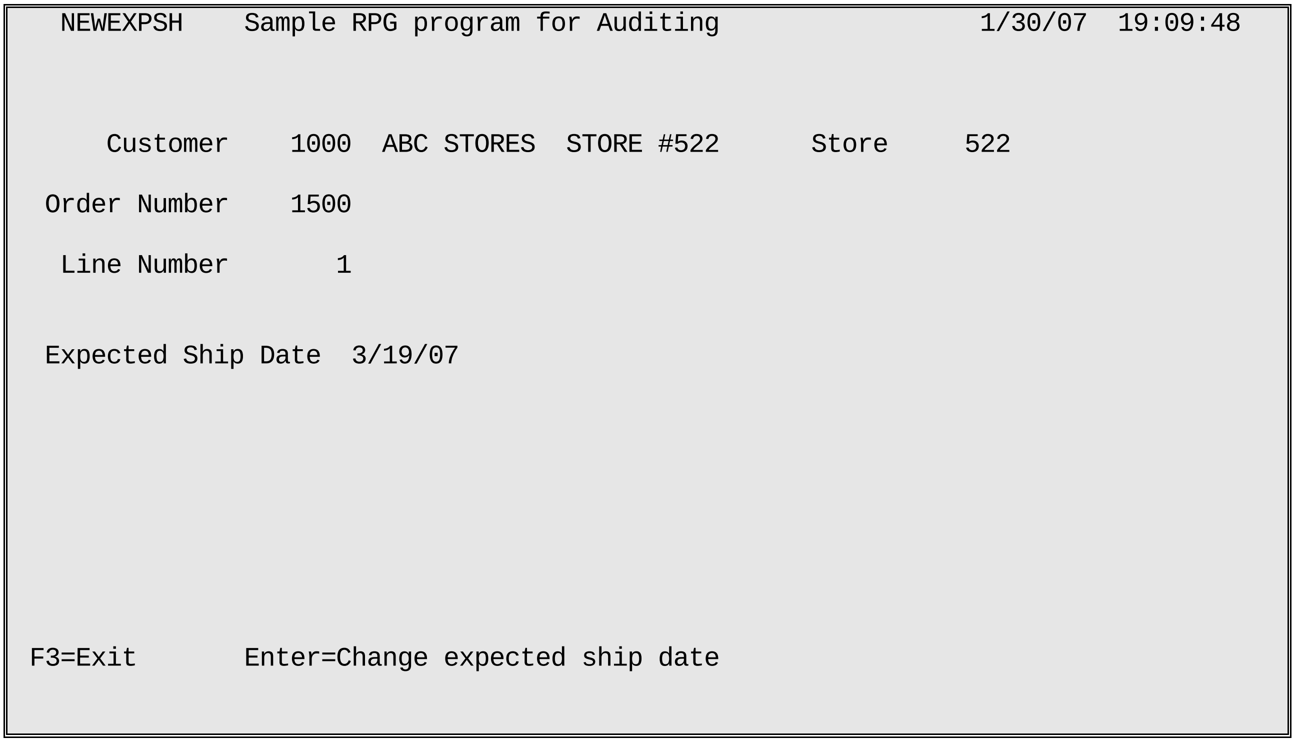
Conditional Auditing with Variable Values

RTPA for RPG provides a very powerful capability to selectively turn auditing on and off based on the contents of variables. RTPA allows you to start, stop, and resume auditing at any point in the execution of the program based on the contents of any variable or combination of variables in the statements being executed.

RTPA conditional auditing with Command Key 8 (F8 at the RTPA main screen) is available before submitting the input source compile for both Full auditing (F10) and advanced auditing (F11).

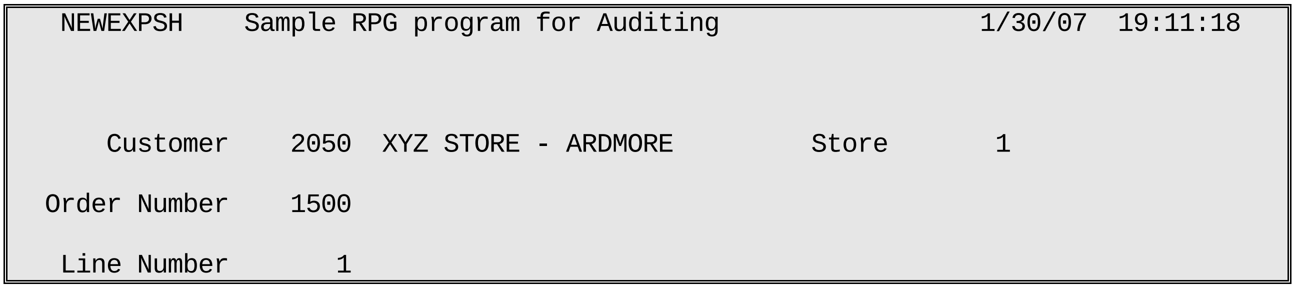
Example of finding a transient error with RTPA

The RTPA for RPG example interactive program NEWEXPSH contains a transient, or intermittent bug that happens only one time in the program. This is when the displayed order number 1500, changes from customer number 1000 (ABC Stores) store 522, to customer number 2050 (XYZ Stores) store 1.

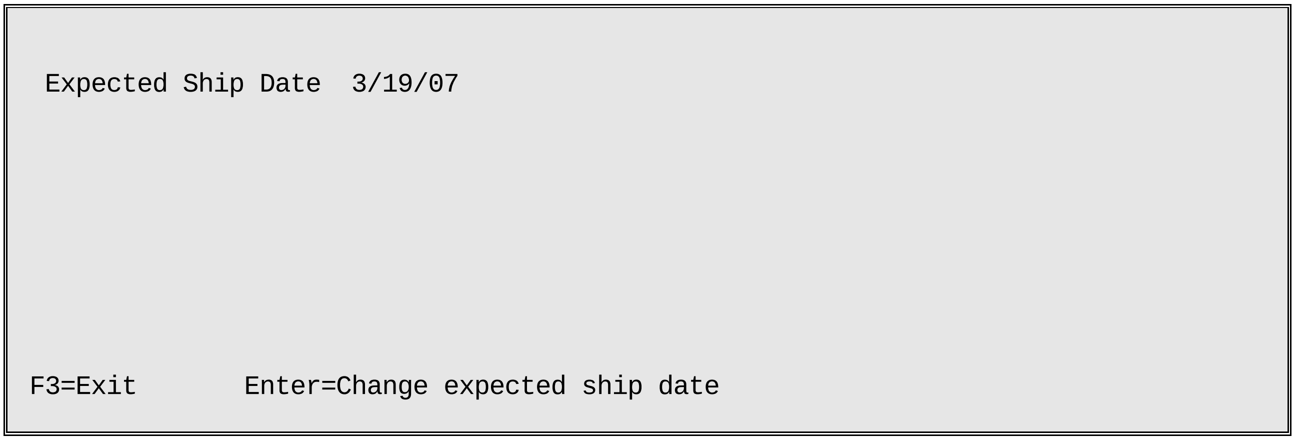


```
  NEWEXPSH    Sample RPG program for Auditing                 1/30/07  19:09:48



     Customer    1000  ABC STORES  STORE #522      Store     522

 Order Number    1500

  Line Number       1


 Expected Ship Date  3/19/07








F3=Exit       Enter=Change expected ship date
```

Figure 4.3 Program NEWEXPSH displaying the correct customer number 1000



```
  NEWEXPSH    Sample RPG program for Auditing                 1/30/07  19:11:18



     Customer    2050  XYZ STORE - ARDMORE         Store       1

 Order Number    1500

  Line Number       1
```



```
 Expected Ship Date  3/19/07








F3=Exit       Enter=Change expected ship date
```

Figure 4.4 Program NEWEXPSH displaying incorrect customer number 2050

The programmer using RTPA for RPG auditing has several effective and simplified methods easily this transient problem, which in a batch program could occur in the millionth order being processed.

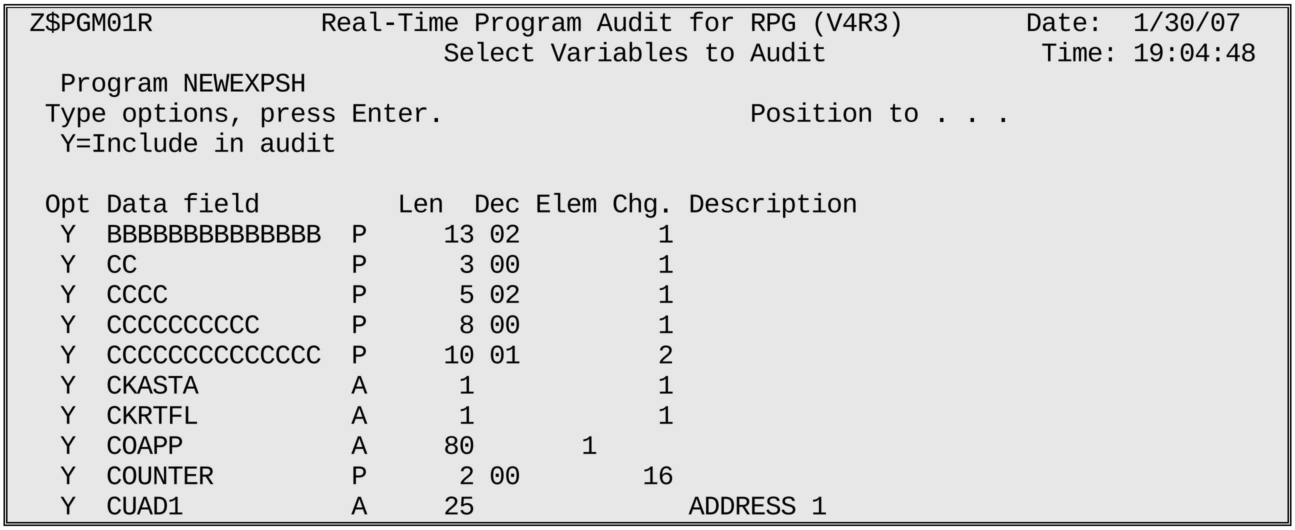
A very simple method is to use the time of the noted error (20:17:08) to review the audit output and search for the first time the customer number 2050 appeared. (Without ever looking at the source program member)

A more powerful RTPA auditing method allows the programmer to start auditing when the customer number is first 2050, and stop auditing when the customer number is no longer 2050. This provides a focused audit of only the error condition and what caused it.

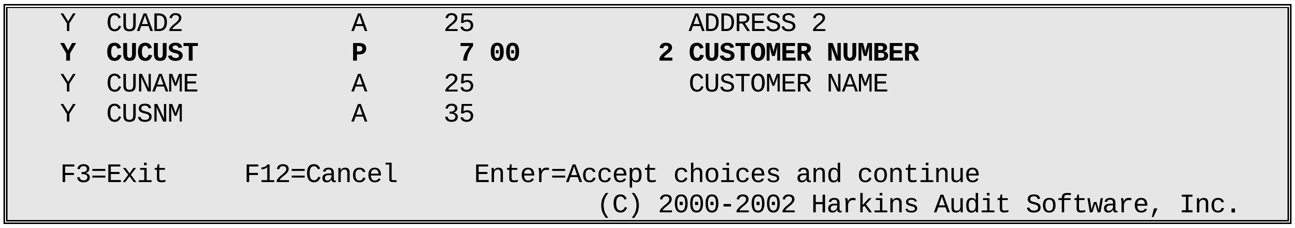
The focused RTPA auditing could also have been turned only when Order # was 1000.

Expanding the NEWEXPSH program with RTPA allows the programmer to display all the variables actually used in the source program, using the F16 command key.

The programmer can then identify the customer number as variable CUCUST.



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  1/30/07
                           Select Variables to Audit              Time: 19:04:48
  Program NEWEXPSH
 Type options, press Enter.                    Position to . . .
  Y=Include in audit

 Opt Data field         Len  Dec Elem Chg. Description
  Y  BBBBBBBBBBBBBB  P     13 02         1
  Y  CC              P      3 00         1
  Y  CCCC            P      5 02         1
  Y  CCCCCCCCCC      P      8 00         1
  Y  CCCCCCCCCCCCCC  P     10 01         2
  Y  CKASTA          A      1            1
  Y  CKRTFL          A      1            1
  Y  COAPP           A     80       1
  Y  COUNTER         P      2 00        16
  Y  CUAD1           A     25              ADDRESS 1
```



```
   Y  CUAD2          A     25              ADDRESS 2
   Y  CUCUST         P      7 00         2 CUSTOMER NUMBER
   Y  CUNAME         A     25              CUSTOMER NAME
   Y  CUSNM          A     35

   F3=Exit     F12=Cancel     Enter=Accept choices and continue
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

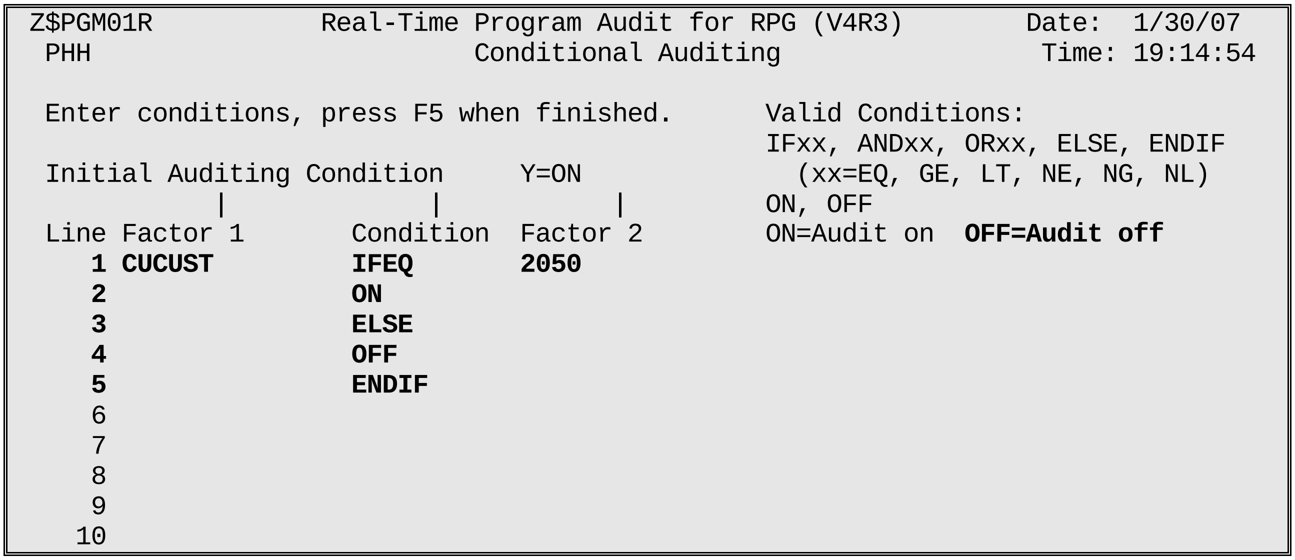
Figure 4.5 The customer number variable is CUCUST

The programmer may now condition RTPA auditing to only audit when variable CUCUST is 2050.

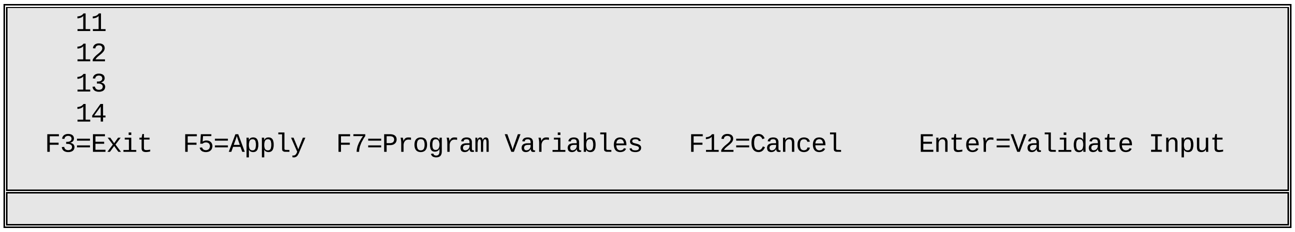
- ❖ **Press F8** to display the `Conditional Auditing` screen.
- ❖ Enter the conditions under which you want the audit to start and stop and **press F5** to apply those conditions.
- ❖ Set the Initial Auditing Condition to ON or OFF. This specifies whether the audit should start when the program starts, or if the audit should only start when the conditions are first met.
- ❖ The Initial Auditing Conditions is blanked (to turn off auditing until the entered condition is met), and the If conditional statement is entered. Command Key 5 is pressed to insert the conditional statement into the expanded RTPA source in library Z$AUDITE.
- ❖ The ON Condition turns RTPA auditing ON
- ❖ The OFF Condition turns RTPA auditing OFF
- ❖ Auditing for only Order number 1500 (field ODORD#) would have been accomplished by changing the If statement to:

```
Line Factor 1       Condition  Factor 2        ON=Audit on  OFF=Audit off
   1 ODORD#        IFEQ       1500
```

- ❖ The IF statement may be a complex statement with multiple IF, AND, OR conditions



```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)       Date:  1/30/07
  PHH                           Conditional Auditing              Time: 19:14:54

  Enter conditions, press F5 when finished.      Valid Conditions:
                                                 IFxx, ANDxx, ORxx, ELSE, ENDIF
  Initial Auditing Condition     Y=ON              (xx=EQ, GE, LT, NE, NG, NL)
             |             |           |         ON, OFF
  Line Factor 1       Condition  Factor 2        ON=Audit on  OFF=Audit off
     1 CUCUST         IFEQ       2050
     2                ON
     3                ELSE
     4                OFF
     5                ENDIF
     6
     7
     8
     9
    10
```

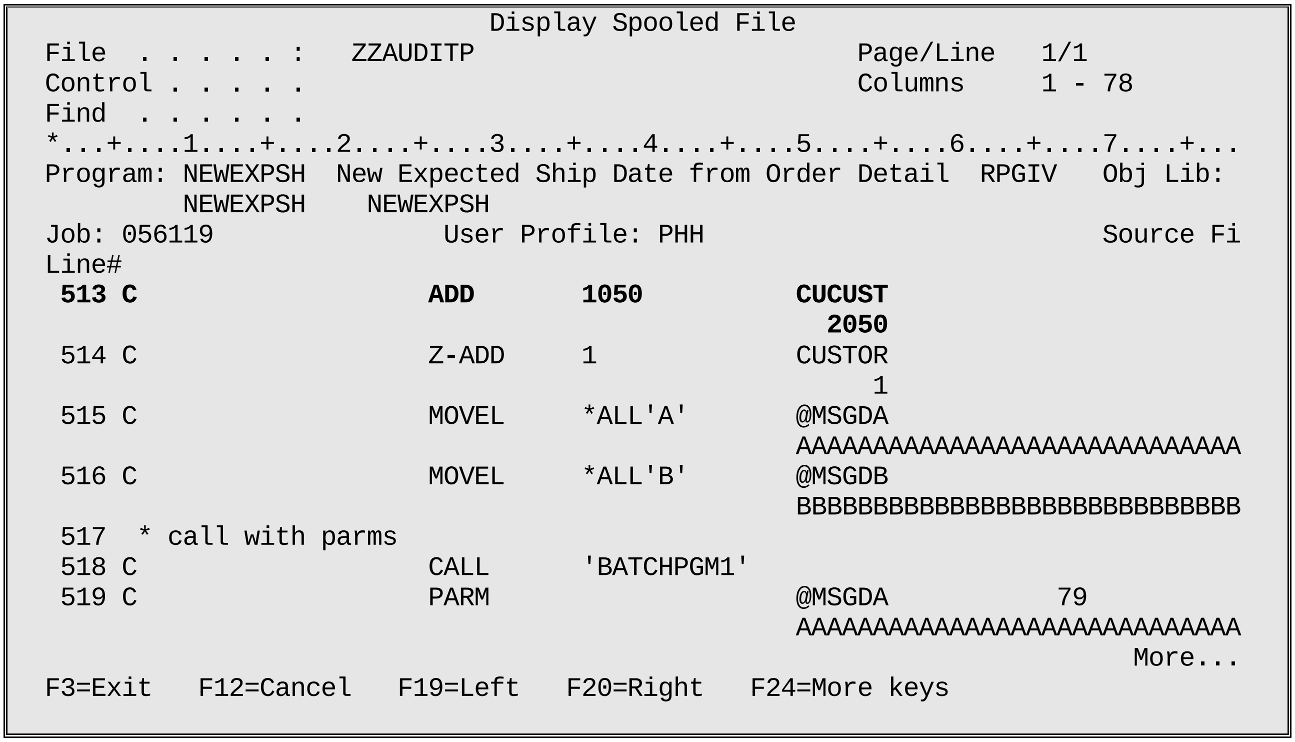


```
  11
  12
  13
  14
F3=Exit  F5=Apply  F7=Program Variables   F12=Cancel     Enter=Validate Input
```

Figure 4.6 Conditional Auditing Screen

- ❖ Press Command key 5 to apply the conditional auditing statements (insert them into the expanded source program), then press command key 10 to submit the program for expansion.

- ❖ The audit output on NEWEXPSH starts at compile statement 513 , when variable CUCUST is 2050 after 1050 is added to the precious contents of the variable CUCUST (which was 1000), and ends when CUCUST is no longer 2050. This provides a very focused audit and identifies the exact source statement causing the transient error. (Without looking at the source member or guessing what happened)



```
                             Display Spooled File
File  . . . . . :   ZZAUDITP                         Page/Line   1/1
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program: NEWEXPSH  New Expected Ship Date from Order Detail  RPGIV   Obj Lib:
         NEWEXPSH    NEWEXPSH
Job: 056119              User Profile: PHH                           Source Fi
Line#
 513 C                  ADD       1050          CUCUST
                                                  2050
 514 C                  Z-ADD     1             CUSTOR
                                                     1
 515 C                  MOVEL     *ALL'A'       @MSGDA
                                                AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 516 C                  MOVEL     *ALL'B'       @MSGDB
                                                BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
 517  * call with parms
 518 C                  CALL      'BATCHPGM1'
 519 C                  PARM                    @MSGDA           79
                                                AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
                                                                      More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

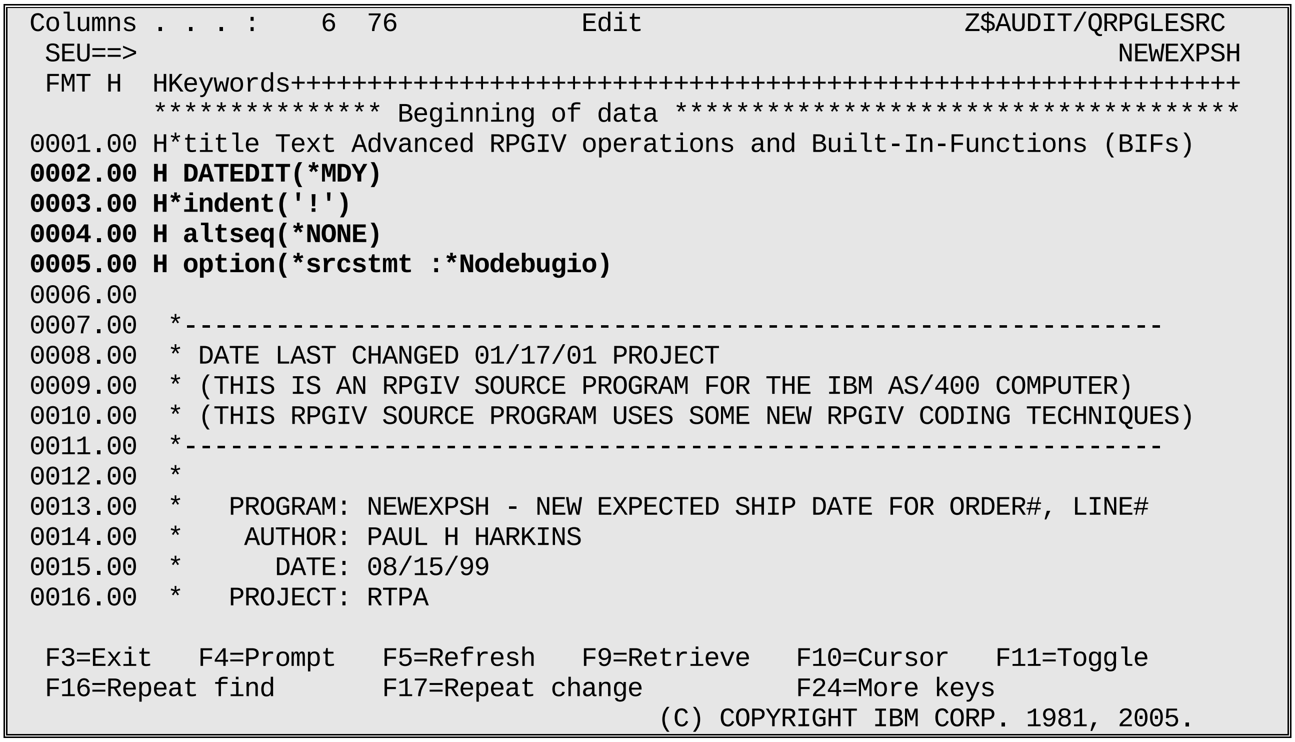
Figure 4.7 The transient error of customer 2050 was caused by source statement 513

Overriding Compile Options

RTPA allows you to override the normal compile options for this Job when creating the audit-enabled program object.

Program source compile overrides for RTPA auditing are at three levels:

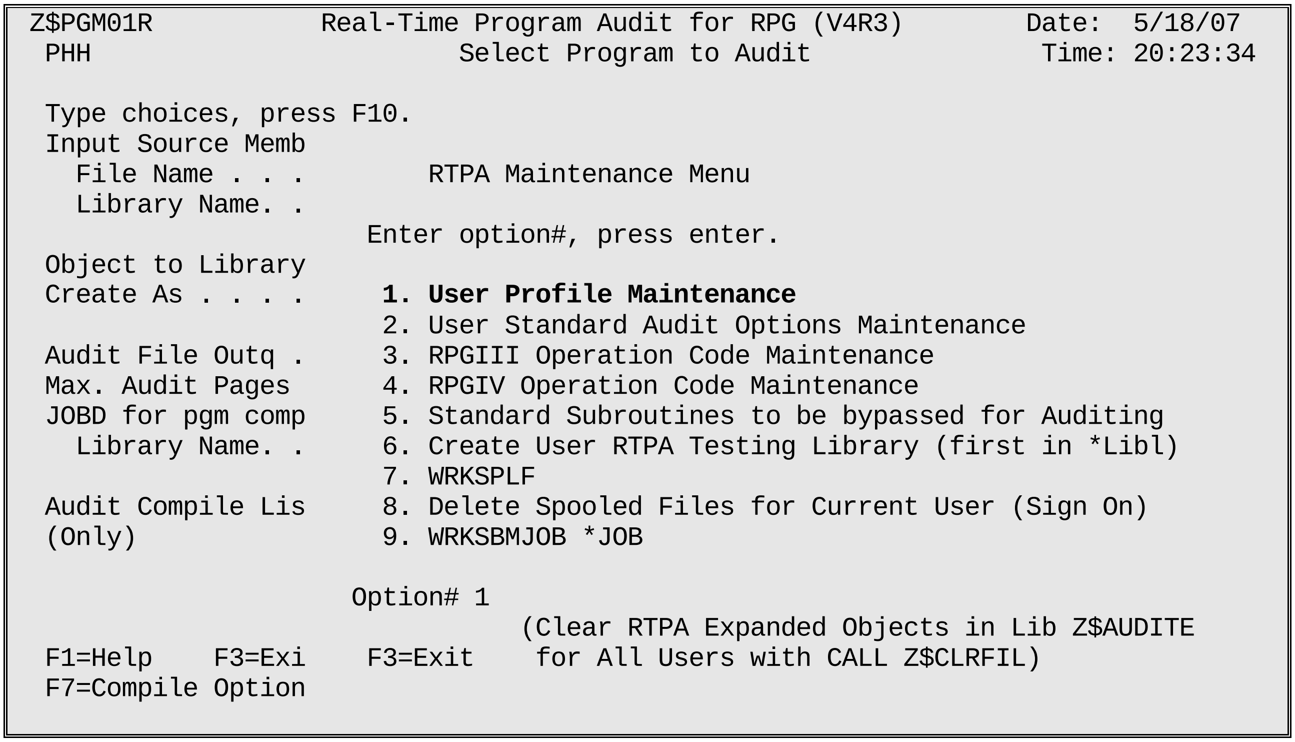
1. The input source program may have Header specifications compile override statements.



```
Columns . . . :    6  76            Edit                       Z$AUDIT/QRPGLESRC
 SEU==>                                                                 NEWEXPSH
 FMT H  HKeywords++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
        *************** Beginning of data ************************************
0001.00 H*title Text Advanced RPGIV operations and Built-In-Functions (BIFs)
0002.00 H DATEDIT(*MDY)
0003.00 H*indent('!')
0004.00 H altseq(*NONE)
0005.00 H option(*srcstmt :*Nodebugio)
0006.00
0007.00  *-------------------------------------------------------------
0008.00  * DATE LAST CHANGED 01/17/01 PROJECT
0009.00  * (THIS IS AN RPGIV SOURCE PROGRAM FOR THE IBM AS/400 COMPUTER)
0010.00  * (THIS RPGIV SOURCE PROGRAM USES SOME NEW RPGIV CODING TECHNIQUES)
0011.00  *-------------------------------------------------------------
0012.00  *
0013.00  *   PROGRAM: NEWEXPSH - NEW EXPECTED SHIP DATE FOR ORDER#, LINE#
0014.00  *    AUTHOR: PAUL H HARKINS
0015.00  *      DATE: 08/15/99
0016.00  *   PROJECT: RTPA

 F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
 F16=Repeat find       F17=Repeat change          F24=More keys
                                           (C) COPYRIGHT IBM CORP. 1981, 2005.
```

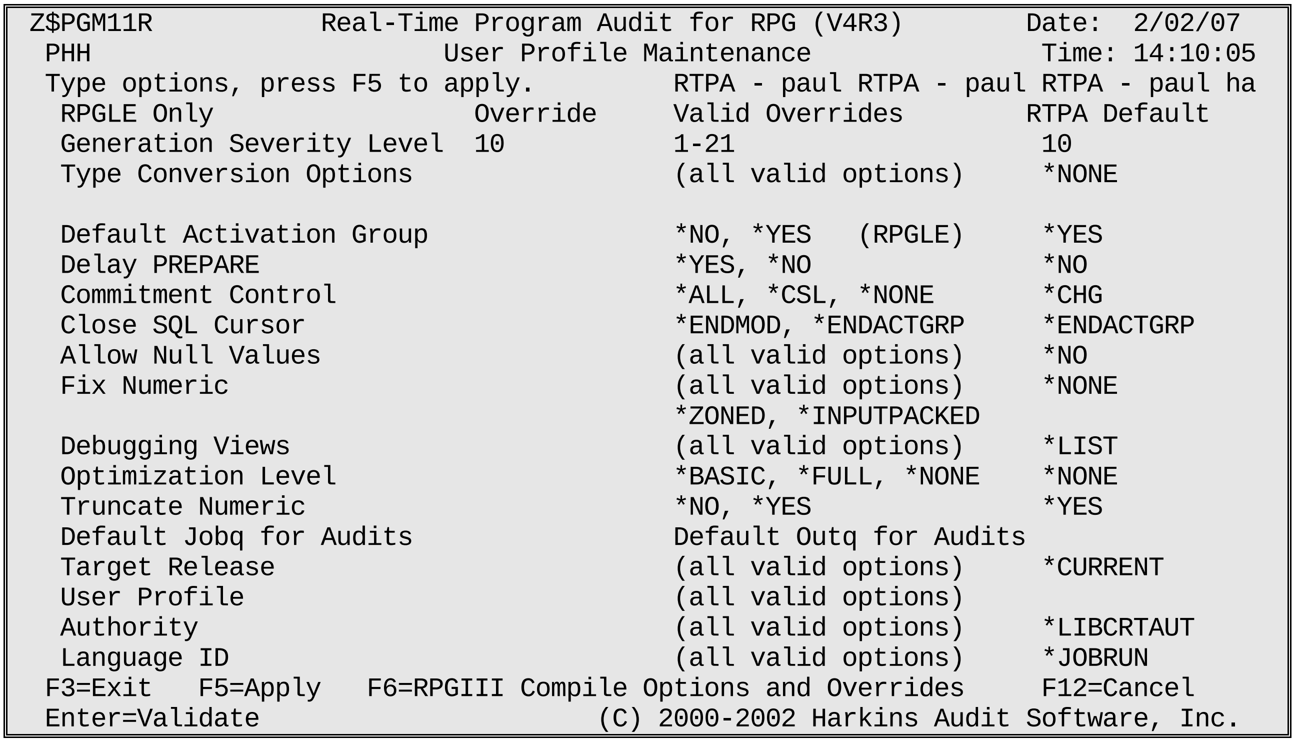
Figure 4.8 RPGLE input source program Header specification compile override statements

2. RTPA User Profile compile override defaults may be used to override the compile options (these compile override options have similar values as using command key 4 to override a normal compile. These RTPA User profile compile options are created dynamically when the user first signs on to RTPA and stored in RTPA file Z$FI01. These User Profile options may be changed using option 1 of the RTPA Menu (command key 9 on the RTPA main screen).



```
Z$PGM01R          Real-Time Program Audit for RPG (V4R3)       Date:  5/18/07
 PHH                           Select Program to Audit           Time: 20:23:34

 Type choices, press F10.
 Input Source Memb
   File Name . . .        RTPA Maintenance Menu
   Library Name. .
                      Enter option#, press enter.
 Object to Library
 Create As . . . .     1. User Profile Maintenance
                       2. User Standard Audit Options Maintenance
 Audit File Outq .     3. RPGIII Operation Code Maintenance
 Max. Audit Pages      4. RPGIV Operation Code Maintenance
 JOBD for pgm comp     5. Standard Subroutines to be bypassed for Auditing
   Library Name. .     6. Create User RTPA Testing Library (first in *Libl)
                       7. WRKSPLF
 Audit Compile Lis     8. Delete Spooled Files for Current User (Sign On)
 (Only)                9. WRKSBMJOB *JOB

                     Option# 1
                                (Clear RTPA Expanded Objects in Lib Z$AUDITE
 F1=Help    F3=Exi    F3=Exit    for All Users with CALL Z$CLRFIL)
 F7=Compile Option
```

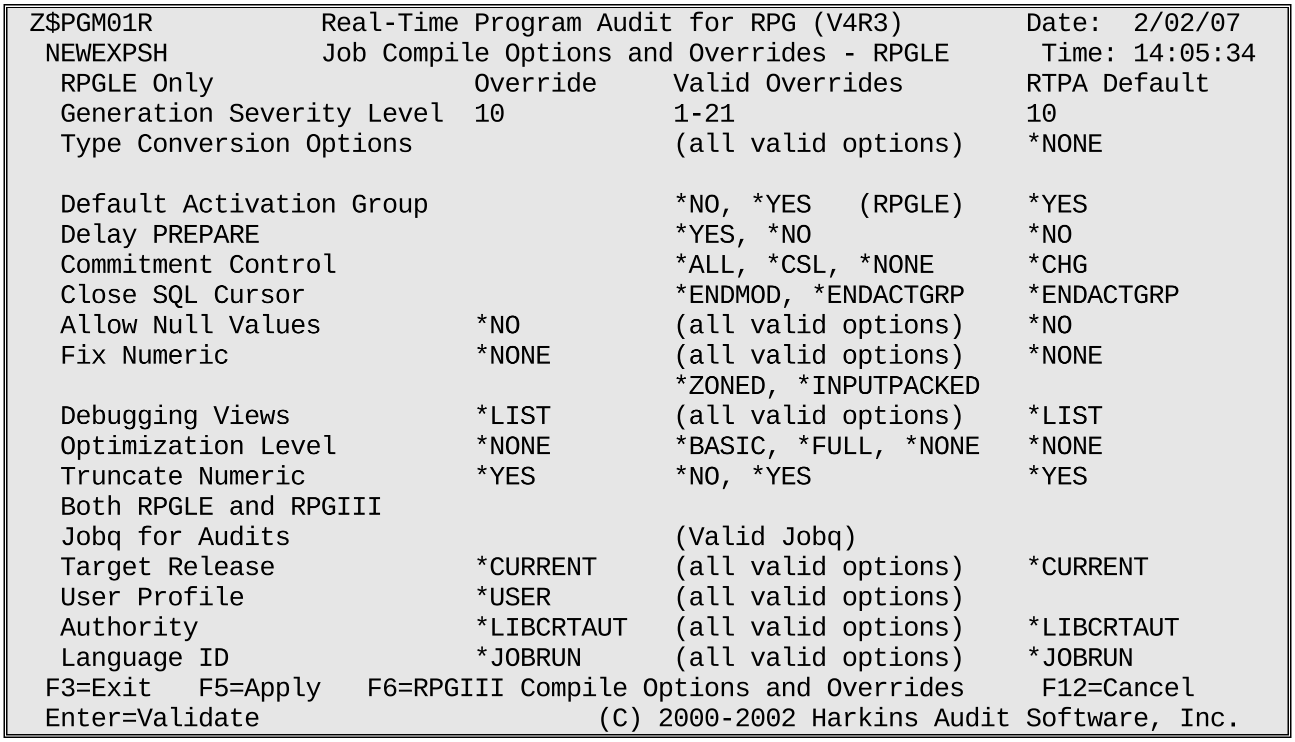
Figure 4.9 Selection of User Profile maintenance to change User compile overrides



```
Z$PGM11R          Real-Time Program Audit for RPG (V4R3)       Date:  2/02/07
 PHH                         User Profile Maintenance            Time: 14:10:05
 Type options, press F5 to apply.         RTPA - paul RTPA - paul RTPA - paul ha
  RPGLE Only                   Override     Valid Overrides        RTPA Default
  Generation Severity Level  10           1-21                     10
  Type Conversion Options                 (all valid options)      *NONE

  Default Activation Group                *NO, *YES   (RPGLE)      *YES
  Delay PREPARE                           *YES, *NO                *NO
  Commitment Control                      *ALL, *CSL, *NONE        *CHG
  Close SQL Cursor                        *ENDMOD, *ENDACTGRP      *ENDACTGRP
  Allow Null Values                       (all valid options)      *NO
  Fix Numeric                             (all valid options)      *NONE
                                          *ZONED, *INPUTPACKED
  Debugging Views                         (all valid options)      *LIST
  Optimization Level                      *BASIC, *FULL, *NONE     *NONE
  Truncate Numeric                        *NO, *YES                *YES
  Default Jobq for Audits                 Default Outq for Audits
  Target Release                          (all valid options)      *CURRENT
  User Profile                            (all valid options)
  Authority                               (all valid options)      *LIBCRTAUT
  Language ID                             (all valid options)      *JOBRUN
 F3=Exit   F5=Apply   F6=RPGIII Compile Options and Overrides     F12=Cancel
 Enter=Validate                      (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.10 RTPA User Profile compile overrides maintenance (for all User compiles)

3. RTPA Job compile overrides are for only this RTPA expansion and are accomplished by pressing Command key 7 on the RTPA main screen.



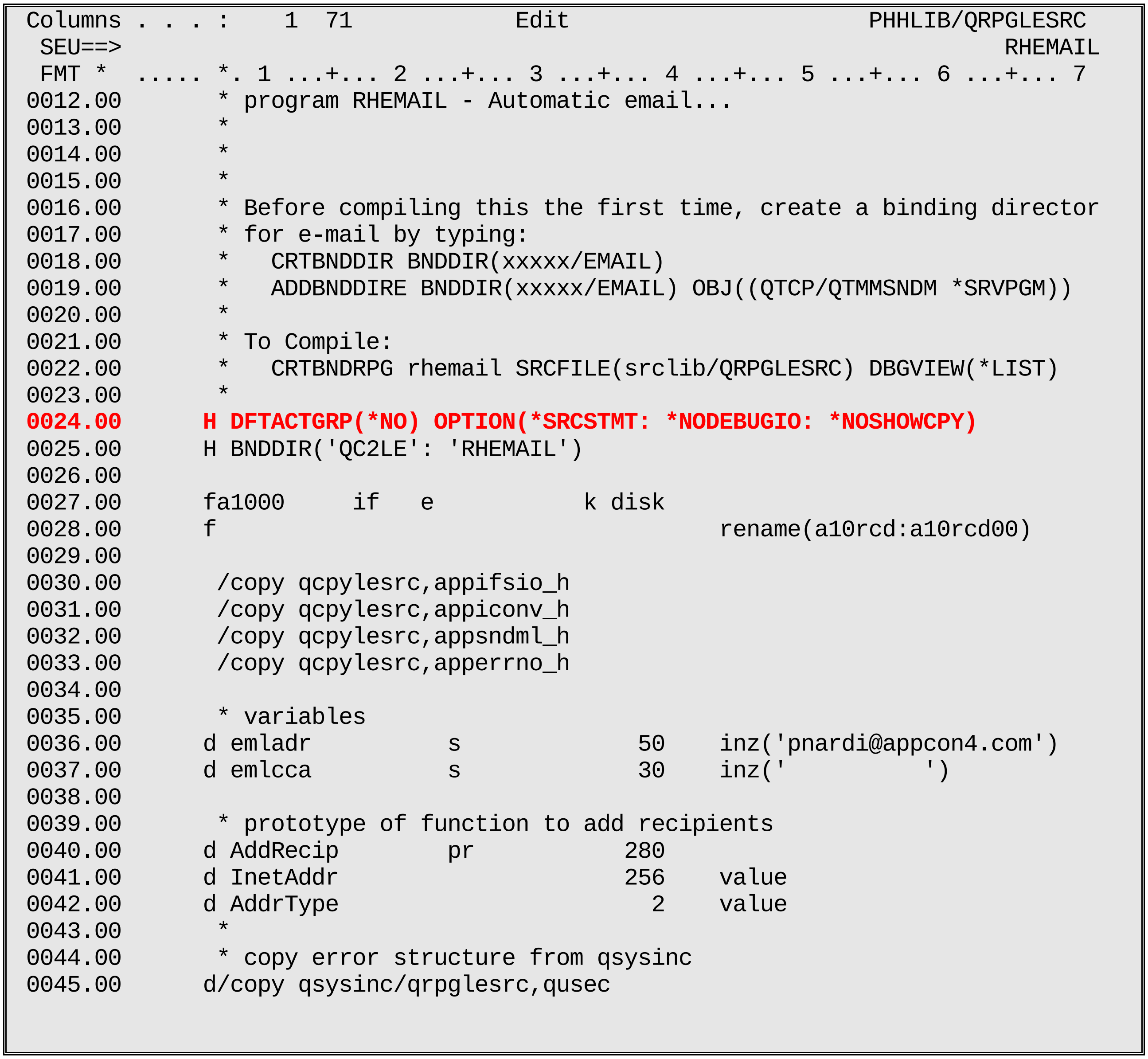
```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  2/02/07
 NEWEXPSH          Job Compile Options and Overrides - RPGLE      Time: 14:05:34
  RPGLE Only                 Override     Valid Overrides        RTPA Default
  Generation Severity Level  10           1-21                   10
  Type Conversion Options                 (all valid options)    *NONE

  Default Activation Group                *NO, *YES   (RPGLE)    *YES
  Delay PREPARE                           *YES, *NO              *NO
  Commitment Control                      *ALL, *CSL, *NONE      *CHG
  Close SQL Cursor                        *ENDMOD, *ENDACTGRP    *ENDACTGRP
  Allow Null Values          *NO          (all valid options)    *NO
  Fix Numeric                *NONE        (all valid options)    *NONE
                                          *ZONED, *INPUTPACKED
  Debugging Views            *LIST        (all valid options)    *LIST
  Optimization Level         *NONE        *BASIC, *FULL, *NONE   *NONE
  Truncate Numeric           *YES         *NO, *YES              *YES
  Both RPGLE and RPGIII
  Jobq for Audits                         (Valid Jobq)
  Target Release             *CURRENT     (all valid options)    *CURRENT
  User Profile               *USER        (all valid options)
  Authority                  *LIBCRTAUT   (all valid options)    *LIBCRTAUT
  Language ID                *JOBRUN      (all valid options)    *JOBRUN
 F3=Exit   F5=Apply   F6=RPGIII Compile Options and Overrides     F12=Cancel
 Enter=Validate                      (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.11 RTPA Job compile overrides with command key 7 on the RTPA main screen

- **Press F5** to apply the Job compile overrides, then F10 or F11 to submit the RTPA expansion

Note: Programmer default compile Options and overrides (for all RTPA expands) are maintained using the Option 1 of the RTPA Maintenance Menu (User Profile Maintenance).

Note: If you are using L date format fields (10 character date) in RPGIII, you must use the `*DATETIME` option for the Type Conversion Option. In RPGIV, the Type Conversion Option must be `*NONE`. To choose the RTPA Default value, leave the Override section blank. You may not enter the default value into the Override column.



```
 Columns . . . :    1  71            Edit                          PHHLIB/QRPGLESRC
  SEU==>                                                                    RHEMAIL
  FMT *  ..... *. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
 0012.00       * program RHEMAIL - Automatic email...
 0013.00       *
 0014.00       *
 0015.00       *
 0016.00       * Before compiling this the first time, create a binding director
 0017.00       * for e-mail by typing:
 0018.00       *   CRTBNDDIR BNDDIR(xxxxx/EMAIL)
 0019.00       *   ADDBNDDIRE BNDDIR(xxxxx/EMAIL) OBJ((QTCP/QTMMSNDM *SRVPGM))
 0020.00       *
 0021.00       * To Compile:
 0022.00       *   CRTBNDRPG rhemail SRCFILE(srclib/QRPGLESRC) DBGVIEW(*LIST)
 0023.00       *
 0024.00      H DFTACTGRP(*NO) OPTION(*SRCSTMT: *NODEBUGIO: *NOSHOWCPY)
 0025.00      H BNDDIR('QC2LE': 'RHEMAIL')
 0026.00
 0027.00      fa1000     if   e           k disk
 0028.00      f                                     rename(a10rcd:a10rcd00)
 0029.00
 0030.00       /copy qcpylesrc,appifsio_h
 0031.00       /copy qcpylesrc,appiconv_h
 0032.00       /copy qcpylesrc,appsndml_h
 0033.00       /copy qcpylesrc,apperrno_h
 0034.00
 0035.00       * variables
 0036.00      d emladr          s             50    inz('pnardi@appcon4.com')
 0037.00      d emlcca          s             30    inz('          ')
 0038.00
 0039.00       * prototype of function to add recipients
 0040.00      d AddRecip        pr           280
 0041.00      d InetAddr                     256    value
 0042.00      d AddrType                       2    value
 0043.00       *
 0044.00       * copy error structure from qsysinc
 0045.00      d/copy qsysinc/qrpglesrc,qusec
```

Figure 4.12 Input source program Compile Options not allowed in RTPA

RTPA for RPG uses the input source program compile spool file to gather information needed to make insert Z$ audit statements. The RTPA for RPG required format for the input RPG source program compile requires a consecutive compile listing sequence number starting with the integer 1. T he following input source program compile options are not allowed by RTPA for RPG and are blanked in the RTPA expanded source program **SRCSTMT NOSHOWCPY INDENT.**
The original input source program is unchanged by RTPA for RPG.

Creating the Expanded Object Program with F10

When you have selected the member to audit, selected any options and determined the library in which to place the audit-enabled program object, RTPA is ready to expand the copied input source [program with Z$ audit statements.

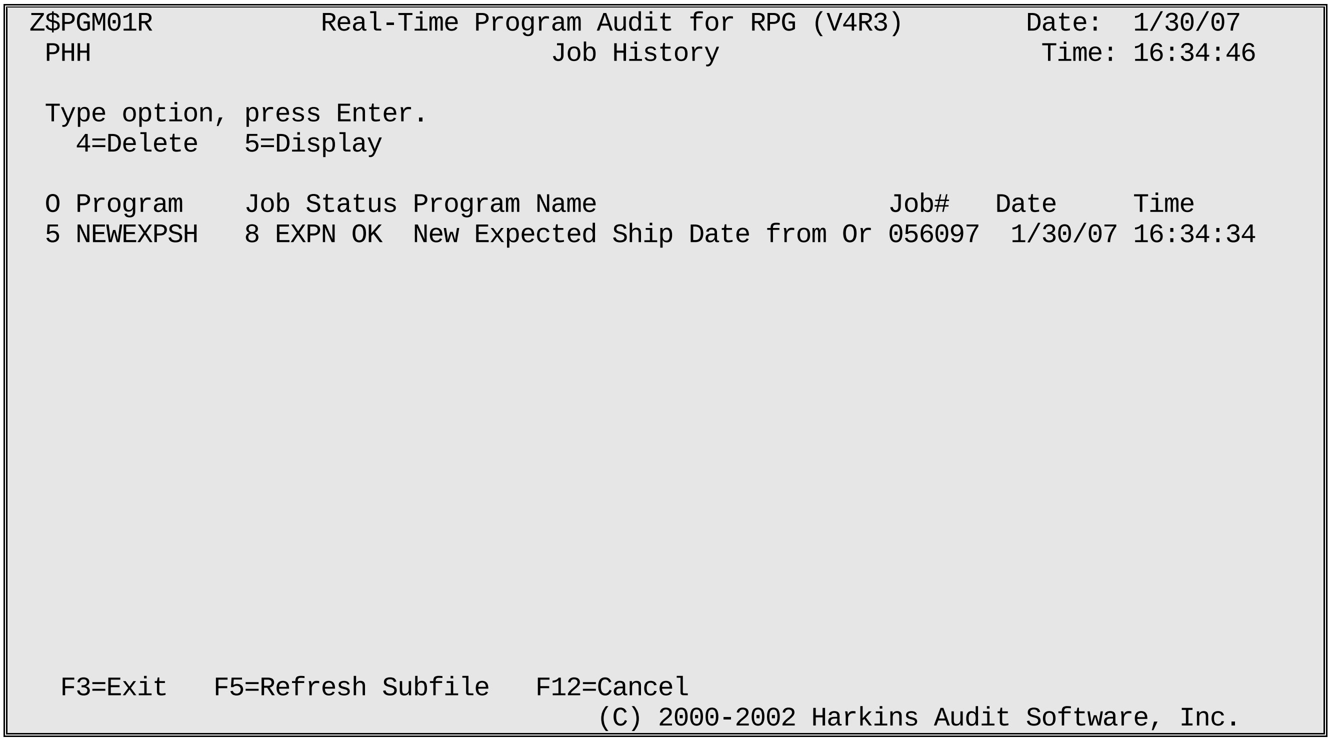
The expanded source program is always a member in a source file in the RTPA library Z$AUDITE, which is the library used by RTPA for all expanded source members, and as the default library for expanded object programs.

- ❖ **To create an audit-enabled program object, press F10 for full RTPA auditing based on the RTPA default or keyed audit options. See Chapter 7 Using Auditing Options for RTPA options control the auditing of expanded programs.**

View Job Status

You can see the status of the job submission on the Job History screen by pressing command 18 on the RTPA main screen.

- ❖ **Press F18** to display the Job History screen as shown in Figure 4.7, where you can see the status of job submission.



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  1/30/07
 PHH                              Job History                       Time: 16:34:46

 Type option, press Enter.
   4=Delete   5=Display

 O Program    Job Status Program Name                      Job#   Date     Time
 5 NEWEXPSH   8 EXPN OK  New Expected Ship Date from Or 056097  1/30/07 16:34:34















  F3=Exit   F5=Refresh Subfile   F12=Cancel
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.13 Programmer Audit Compiles by submitted Job# Screen (for Today)

RTPA Expansion Status Codes

Status Code	Description
1	Compiling Original Source
2	Good Compile of Original Source
3	**Bad Compile of Original Source**
4	Expanding Source
5	Good Expansion of Source
6	Bad Expansion of Source
7	Compiling Expanded Source
8	**Good Compile of Expanded Source**
9	**Error Compiling Expanded Source**

```
O Program    Job Status Program Name                   Job#   Date     Time
5 NEWEXPSH   8 EXPN OK  New Expected Ship Date from Or 056097  1/30/07 16:34:34
```

RTPA **Status code 8 EXPN OK** means that the expanded RTPA source program with Z$ audit statements in library Z$AUDITE has compile correctly and the expanded object program may now be used to create program as the program executes.

Note: **Error Code 3** occurs if the input source program will not compile. This may be because of errors in the input source program or a library list problem..

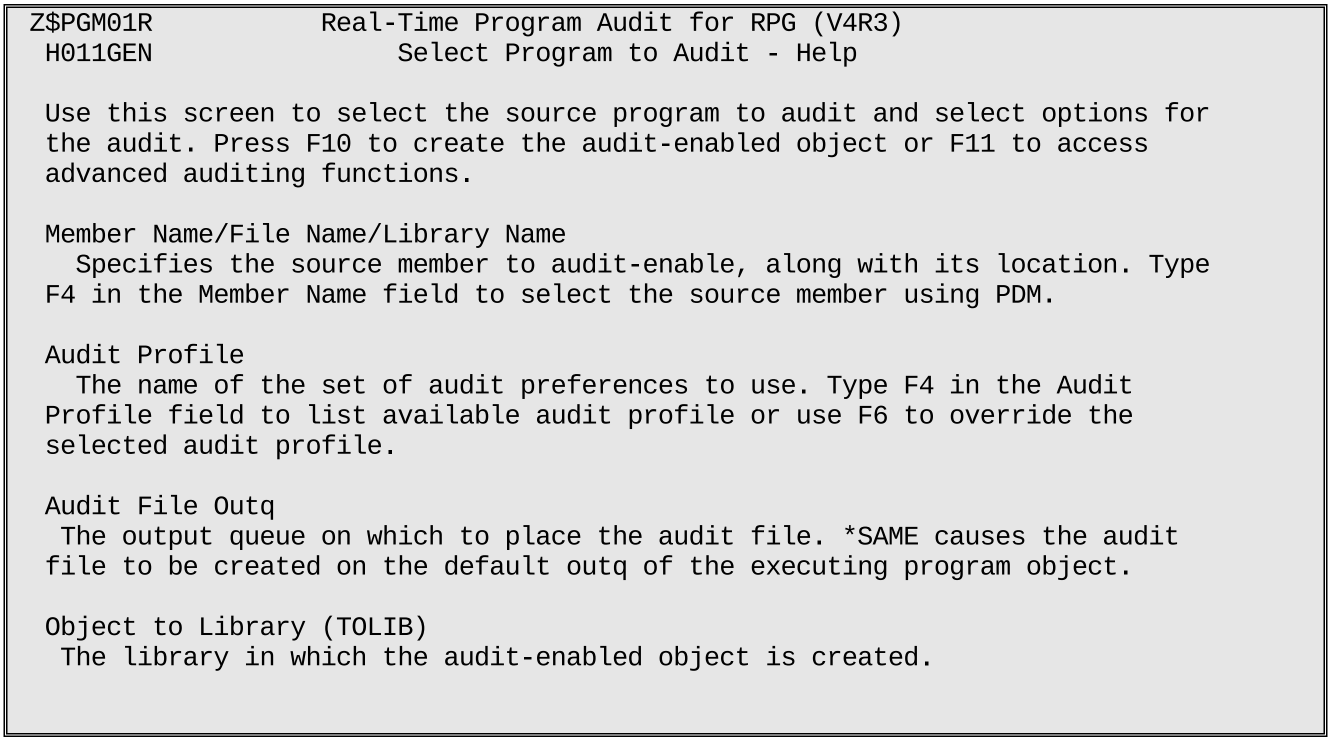
Note: **Error Code 9** can occur if the expansion stage of the source makes the code too large for the compiler. See **Chapter 8: Auditing Very Large RPG Programs** to see how to handle Error Code 9. Error code 9 may also occur if RTPA incorrectly inserts Z$ audit statements. This error may be corrected by commenting the invalid Z$ audit statement in the source member in library Z$AUDITE and recompiling the source member from library Z$AUDITE.

Note: Copybook input source programs should not be expanded with RTPA Z$ audit statements, as the expanded copybook source is put into library Z$AUDITE. Copybook source copied into RPG source programs with the /COPY statement are audited unless excluded with RTPA an option.

Built-In Help

Real-Time Program Audit provides full cursor-sensitive online help text support. RTPA online help allows the user to review detailed information about the screens, important screen fields and command keys.

- To get online help, place the cursor at the field where you want help information and then **press F1**.



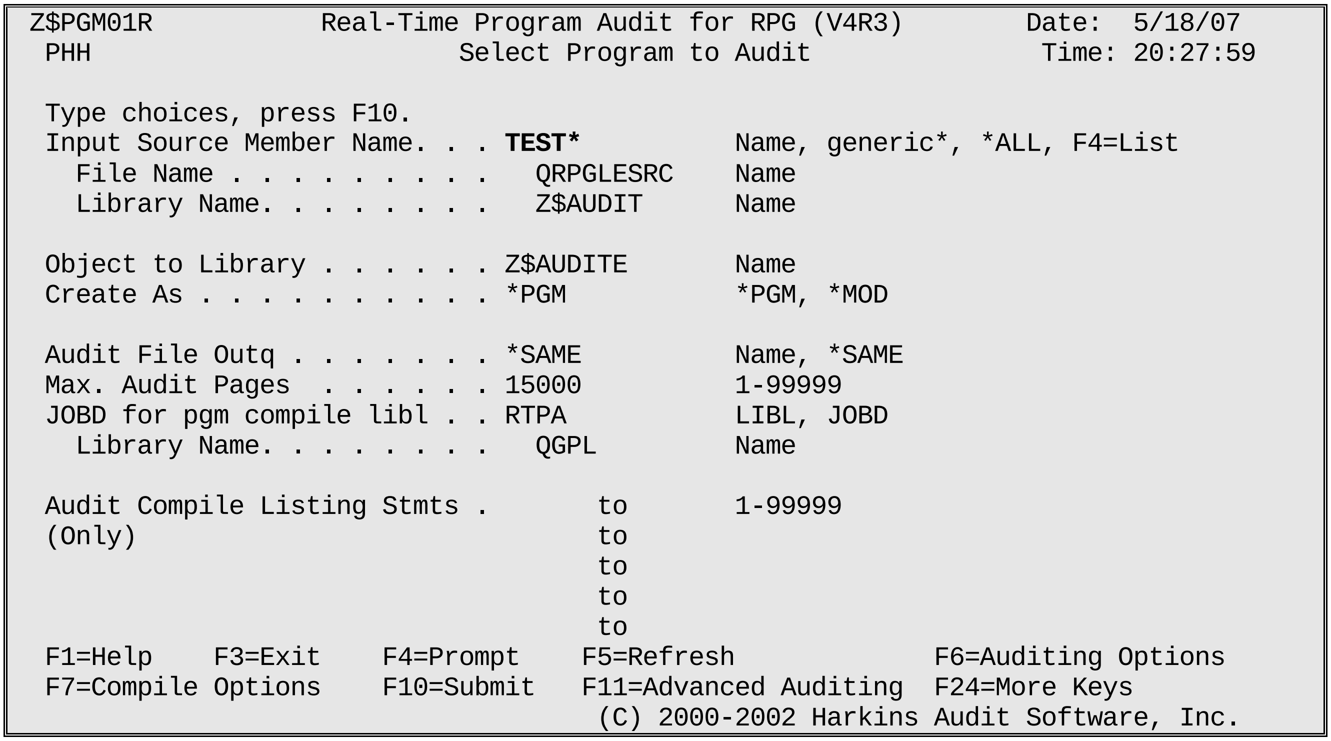
```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)
 H011GEN                Select Program to Audit - Help

 Use this screen to select the source program to audit and select options for
 the audit. Press F10 to create the audit-enabled object or F11 to access
 advanced auditing functions.

 Member Name/File Name/Library Name
   Specifies the source member to audit-enable, along with its location. Type
 F4 in the Member Name field to select the source member using PDM.

 Audit Profile
   The name of the set of audit preferences to use. Type F4 in the Audit
 Profile field to list available audit profile or use F6 to override the
 selected audit profile.

 Audit File Outq
  The output queue on which to place the audit file. *SAME causes the audit
 file to be created on the default outq of the executing program object.

 Object to Library (TOLIB)
  The library in which the audit-enabled object is created.
```

Figure 4.14 RTPA Help Text

Selecting multiple source members (Mass compiles)

A generic program group of source programs in a source file (an * after the program name), or all the source programs in a source file (*ALL in the program name), may be expanded with RTPA audit statements, using notation similar to that used in PDM (Program Development Manager).

In the following illustration, all of the programs in beginning with TEST in the source file QRPGLESRC will be expanded with RTPA audit statements, and the expanded object programs will be placed in library Z$AUDITE. The source type of the program (eg. RPGLE SQLRPGLE), and the Create as option are used in the expansion. Together with the program audit options (Command Key 6, and the compiler overrides).



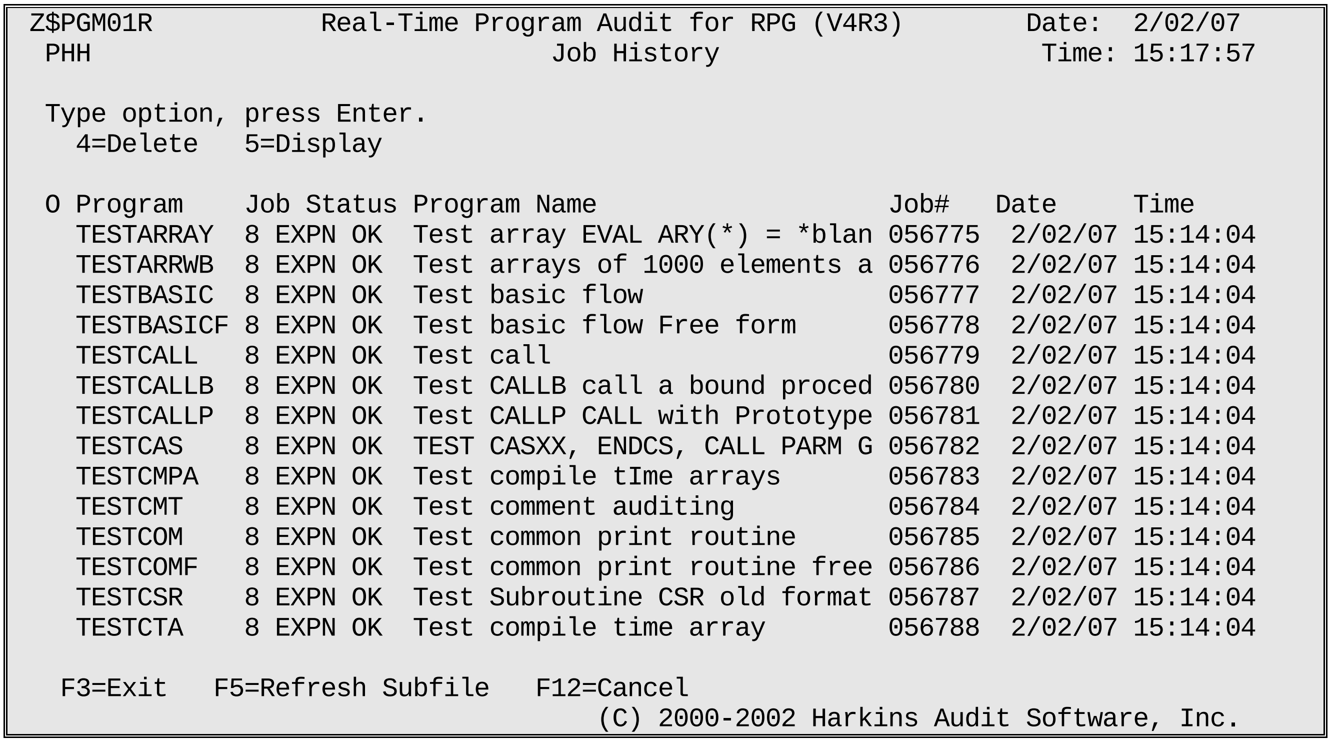
```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)       Date:  5/18/07
 PHH                         Select Program to Audit             Time: 20:27:59

 Type choices, press F10.
 Input Source Member Name. . . TEST*          Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QRPGLESRC    Name
   Library Name. . . . . . . .   Z$AUDIT      Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 15000          1-99999
 JOBD for pgm compile libl . . RTPA           LIBL, JOBD
   Library Name. . . . . . . .   QGPL         Name

 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F1=Help    F3=Exit    F4=Prompt    F5=Refresh             F6=Auditing Options
 F7=Compile Options    F10=Submit   F11=Advanced Auditing  F24=More Keys
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.15 Selection of RTPA of all source programs starting with TEST

Press command key 10 to submit programs for mass expansion

The message: **Member TEST*   submitted. Press F18 to see status. Mass compiles** indicates that the programs have been submitted for RTPA expansion.

Press command key 18 to review the expansion status of the submitted programs



```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  2/02/07
  PHH                              Job History                     Time: 15:17:57

  Type option, press Enter.
    4=Delete    5=Display

  O Program    Job Status Program Name                   Job#   Date     Time
    TESTARRAY  8 EXPN OK  Test array EVAL ARY(*) = *blan 056775  2/02/07 15:14:04
    TESTARRWB  8 EXPN OK  Test arrays of 1000 elements a 056776  2/02/07 15:14:04
    TESTBASIC  8 EXPN OK  Test basic flow                056777  2/02/07 15:14:04
    TESTBASICF 8 EXPN OK  Test basic flow Free form      056778  2/02/07 15:14:04
    TESTCALL   8 EXPN OK  Test call                      056779  2/02/07 15:14:04
    TESTCALLB  8 EXPN OK  Test CALLB call a bound proced 056780  2/02/07 15:14:04
    TESTCALLP  8 EXPN OK  Test CALLP CALL with Prototype 056781  2/02/07 15:14:04
    TESTCAS    8 EXPN OK  TEST CASXX, ENDCS, CALL PARM G 056782  2/02/07 15:14:04
    TESTCMPA   8 EXPN OK  Test compile tIme arrays       056783  2/02/07 15:14:04
    TESTCMT    8 EXPN OK  Test comment auditing          056784  2/02/07 15:14:04
    TESTCOM    8 EXPN OK  Test common print routine      056785  2/02/07 15:14:04
    TESTCOMF   8 EXPN OK  Test common print routine free 056786  2/02/07 15:14:04
    TESTCSR    8 EXPN OK  Test Subroutine CSR old format 056787  2/02/07 15:14:04
    TESTCTA    8 EXPN OK  Test compile time array        056788  2/02/07 15:14:04

   F3=Exit   F5=Refresh Subfile   F12=Cancel
                                      (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.16 RTPA expansion of all source programs starting with TEST

RTPA **Status code 8 EXPN OK** means that the expanded RTPA source programs with Z$ audit statements in library Z$AUDITE has compile correctly and the expanded object program may now be used to create program as the program executes.

Note: Copybook input source programs should not be expanded with RTPA Z$ audit statements, as the expanded copybook source is put into library Z$AUDITE. Copybook source copied into RPG source programs with the /COPY statement are audited unless excluded with RTPA an option.

Instant RTPA Program Auditing with the iRTPA command

The **iRTPA** command (Instant RTPA) provides the User with a powerful shortcut to audit enable an RPG program or many programs.

The User may simply key iRTPA on the command line and press the Enter key to submit the last audit enabled program for another RTPA expansion,. The iRTPA command bypasses the RTPA main selection screen, and displays the RTPA Job Summary screen showing the submitted program or programs (for generic programs).

This iRTPA command is very useful and quick when testing the same program (or programs) with full RTPA auditing.

The blindly fast current System i processors, and the forthcoming System i Power6 and Power7 processors, allow for virtually instant RTPA expansion of a program or selected generic programs, and thus provide for virtually instant full electronic program auditing of all programs in an application.

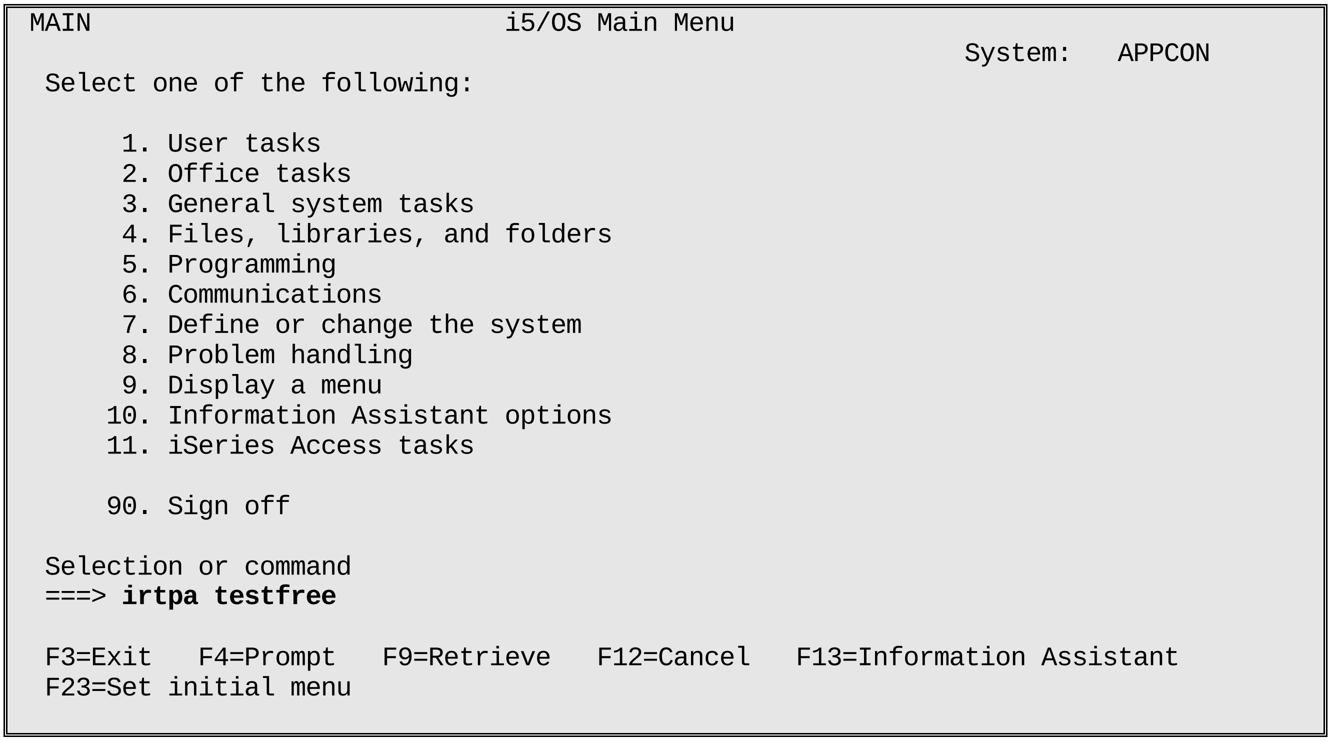
Like the RTPA Command, the iRTPA command optionally allows the program Name (or generic program name) to also be keyed.

`irtpa testfree  (expands RPG source program TESTFREE)`

`irtpa test*     (expands all RPG source program beginning with TEST)`

`irtpa *ALL      (expands all RPG source program in the source file`

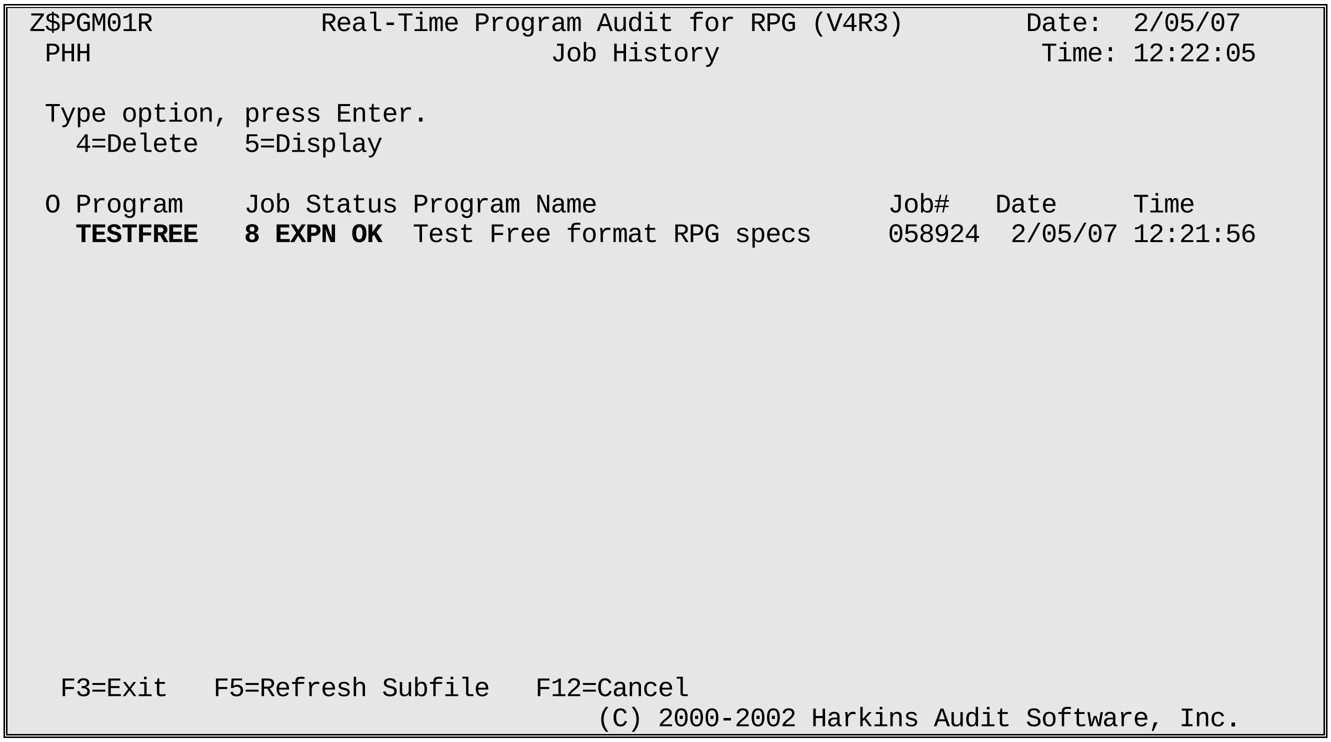
The expanded object library and other RTPA main screen defaults from the last RTPA expand for the User are used, together with the RTPA User profile defaults.



```
 MAIN                           i5/OS Main Menu
                                                              System:   APPCON
  Select one of the following:

       1. User tasks
       2. Office tasks
       3. General system tasks
       4. Files, libraries, and folders
       5. Programming
       6. Communications
       7. Define or change the system
       8. Problem handling
       9. Display a menu
      10. Information Assistant options
      11. iSeries Access tasks

      90. Sign off

  Selection or command
  ===> irtpa testfree

  F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
  F23=Set initial menu
```

Figure 4.17 Selection of the iRTPA command with a keyed program name to be expanded

The Enter key is pressed, and the TESTFREE program is submitted for RTPA expansion.
The RTPA Job History Summary screen is immediately displayed (bypassing the RTPA main selection screen), and the status of the submitted job is shown.

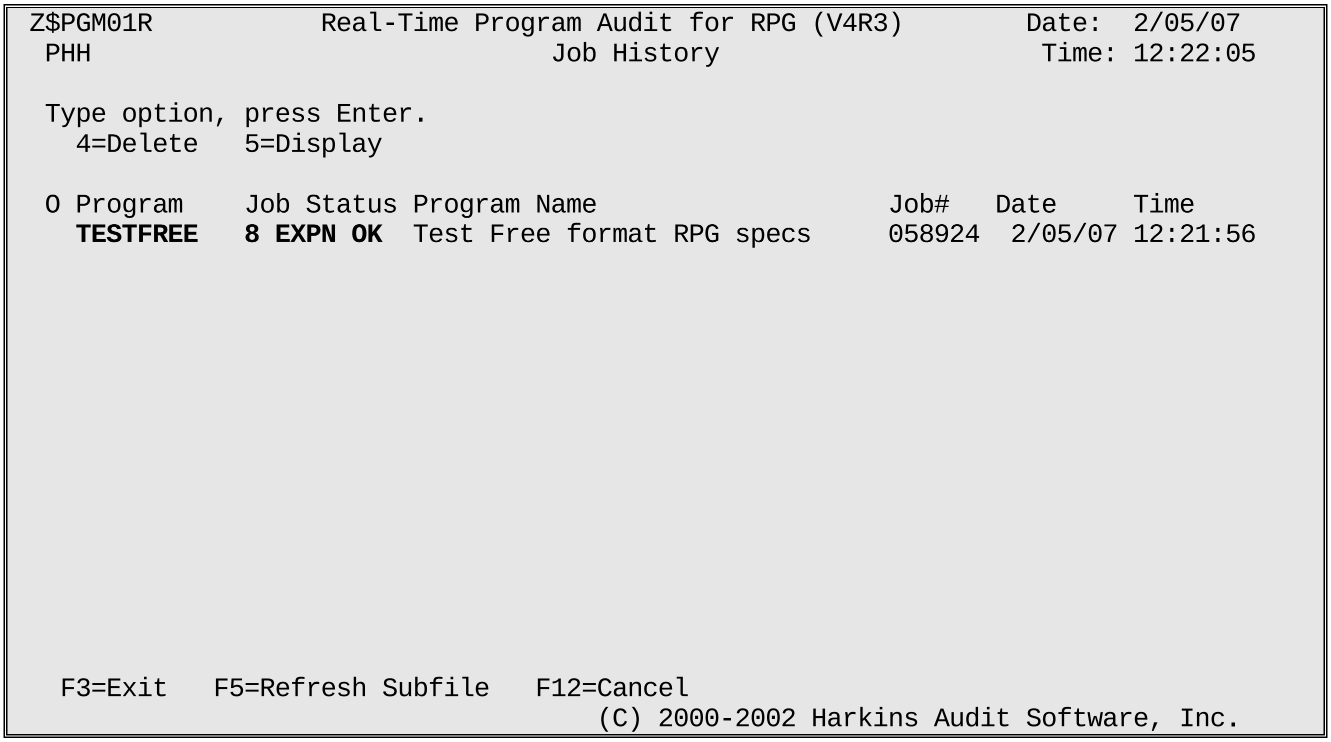


```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  2/05/07
 PHH                              Job History                      Time: 12:22:05

 Type option, press Enter.
   4=Delete    5=Display

 O Program    Job Status Program Name                   Job#   Date     Time
   TESTFREE   8 EXPN OK  Test Free format RPG specs     058924  2/05/07 12:21:56










  F3=Exit   F5=Refresh Subfile   F12=Cancel
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.18 Display of the submitted program or programs for RTPA expansion for the User

The RTPA current expansion status may be displayed by pressing the enter key or the command 5 key.

The RTPA Expand Status is 1 for the submit of the input RPG source program for a compile.
The RTPA Expand **Status is 8 for the successful compile of the expanded source. Ready for auditing**.



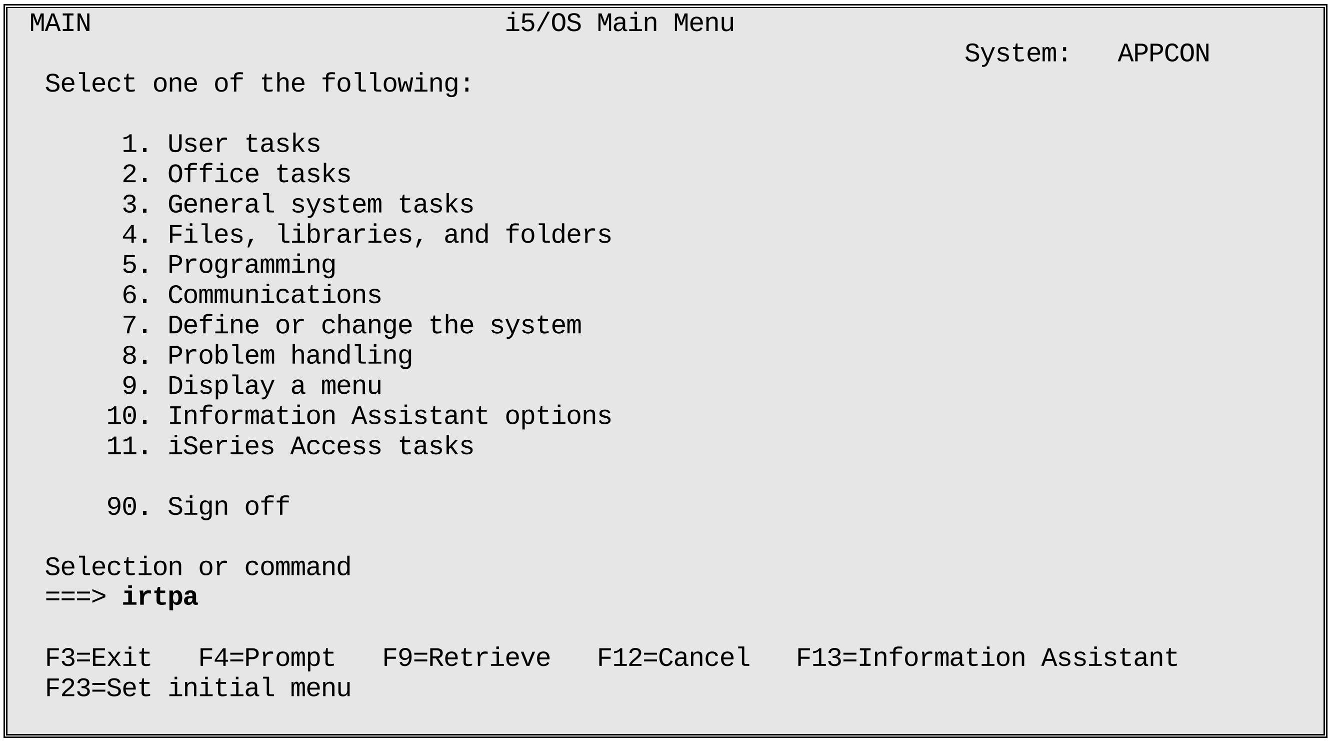
```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  2/05/07
 PHH                               Job History                     Time: 12:22:05

 Type option, press Enter.
   4=Delete   5=Display

 O Program    Job Status Program Name                   Job#   Date     Time
   TESTFREE   8 EXPN OK  Test Free format RPG specs     058924  2/05/07 12:21:56
















  F3=Exit   F5=Refresh Subfile   F12=Cancel
                                      (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.19 Display of the submitted program or programs for RTPA expansion for the User

Like the RTPA Command, the iRTPA command alone may be keyed and the enter key pressed to submit the RPG source program last successfully expanded by the User for an expansion with audit statements. keyed.

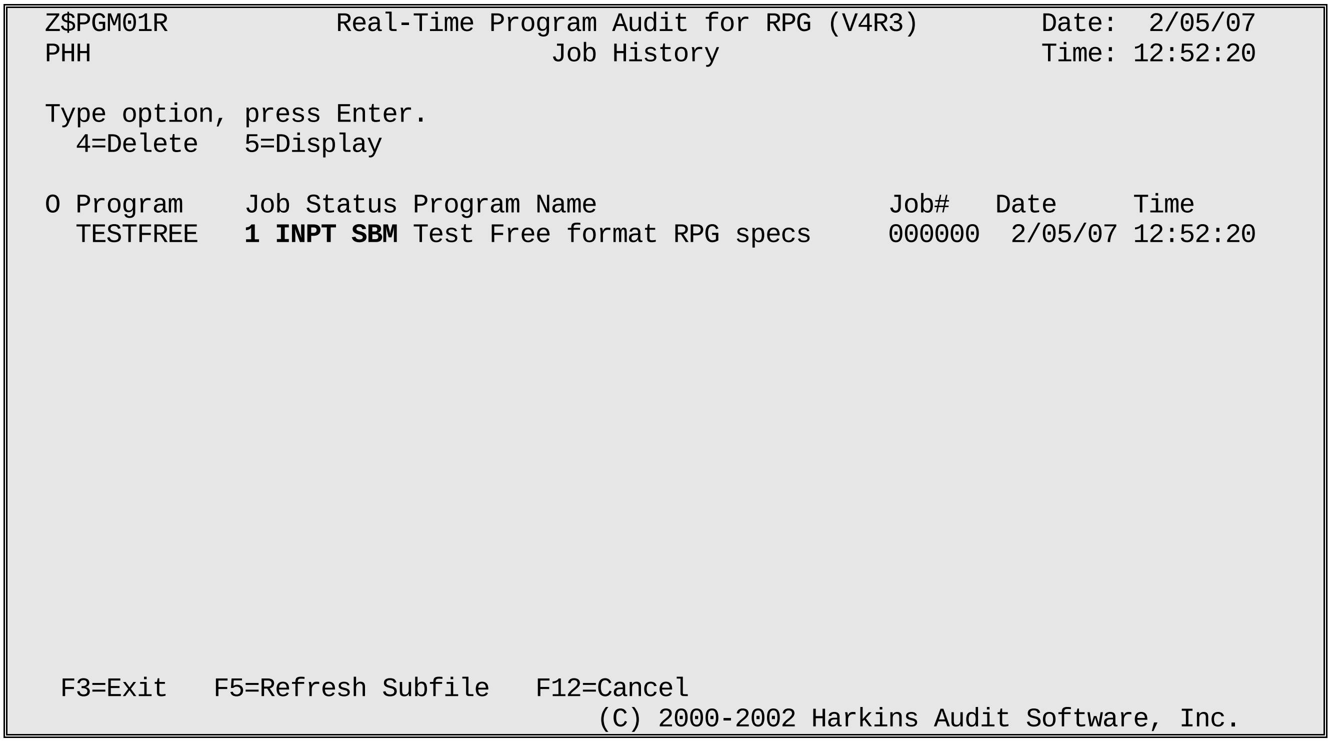
irtpa

The expanded object library and other RTPA main screen defaults from the last RTPA (or iRTPA) expand for the User are used, together with the RTPA User profile defaults.



```
MAIN                           i5/OS Main Menu
                                                              System:   APPCON
 Select one of the following:

      1. User tasks
      2. Office tasks
      3. General system tasks
      4. Files, libraries, and folders
      5. Programming
      6. Communications
      7. Define or change the system
      8. Problem handling
      9. Display a menu
     10. Information Assistant options
     11. iSeries Access tasks

     90. Sign off

 Selection or command
 ===> irtpa

 F3=Exit   F4=Prompt   F9=Retrieve   F12=Cancel   F13=Information Assistant
 F23=Set initial menu
```

Figure 4.20 Selection of the iRTPA command without a keyed program name (Use the last successfully expanded RPG source program name for the User)



```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)       Date:  2/05/07
 PHH                              Job History                    Time: 12:52:20

 Type option, press Enter.
   4=Delete   5=Display

 O Program    Job Status Program Name                  Job#   Date     Time
   TESTFREE   1 INPT SBM Test Free format RPG specs     000000  2/05/07 12:52:20














  F3=Exit   F5=Refresh Subfile   F12=Cancel
                                    (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.21 iRTPA command without a keyed program name submits the last source name expanded

iRTPA TEST*   expands all RPG programs in the source file and library starting with TEST .

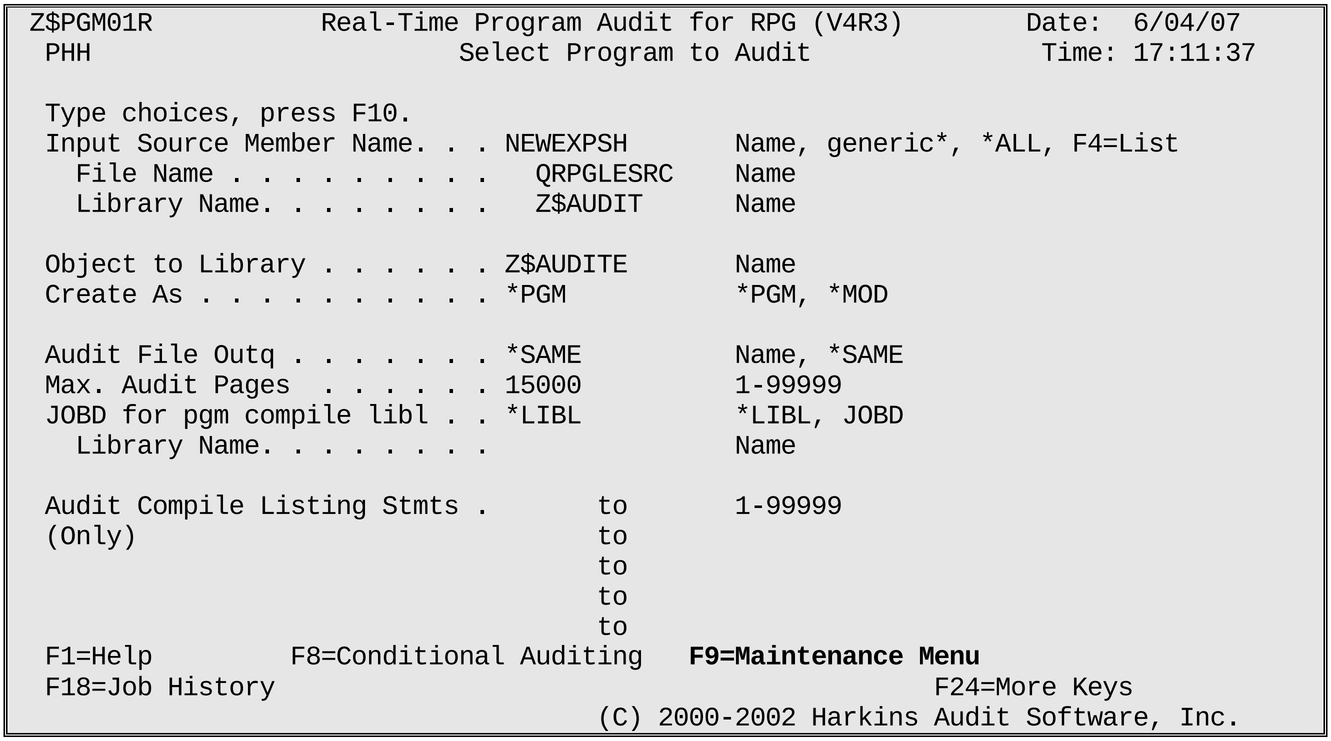
iRTPA *ALL expands ALL RPG programs in the source file and library. Expanding all the RPG source programs in an applications effectively provided RTPA program auditing for all RPG application programs executed at all levels of execution in the program call stack.

Using the RTPA Maintenance Menu to manage RTPA

The RTPA Menu provides the Company using RTPA auditing and each User using RTPA auditing with useful features for managing auditing.

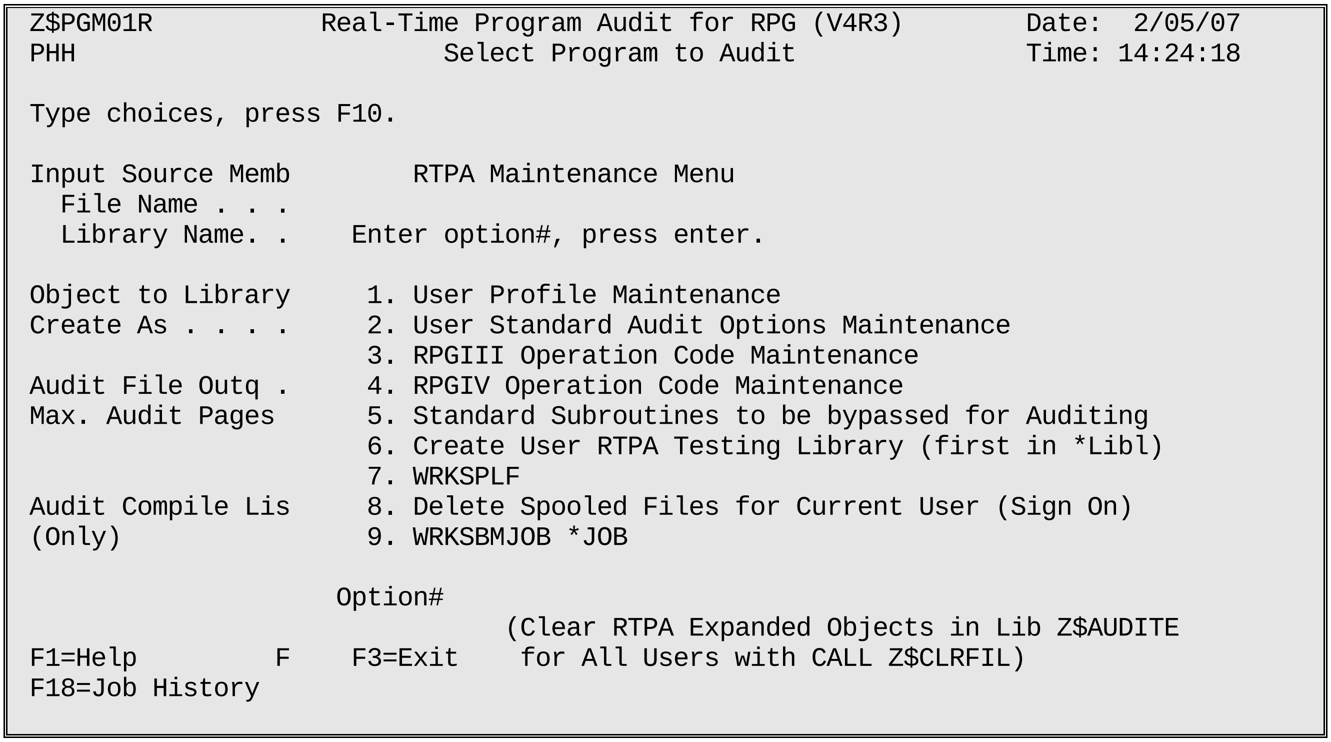
Note – Command Key 24 on the Main RTPA selection screen may be pressed to toggle the RTPA command key features at the bottom of the screen.

The RTPA Maintenance Menu is accessed by pressing command key 9 at the RTPA main selection screen.



```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  6/04/07
  PHH                        Select Program to Audit               Time: 17:11:37

  Type choices, press F10.
  Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
    File Name . . . . . . . . .   QRPGLESRC    Name
    Library Name. . . . . . . .   Z$AUDIT      Name

  Object to Library . . . . . . Z$AUDITE       Name
  Create As . . . . . . . . . . *PGM           *PGM, *MOD

  Audit File Outq . . . . . . . *SAME          Name, *SAME
  Max. Audit Pages  . . . . . . 15000          1-99999
  JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
    Library Name. . . . . . . .                Name

  Audit Compile Listing Stmts .       to       1-99999
  (Only)                              to
                                      to
                                      to
                                      to
  F1=Help         F8=Conditional Auditing   F9=Maintenance Menu
  F18=Job History                                          F24=More Keys
                                      (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.22 Selecting the RTPA Maintenance Menu from the RTPA main selection screen



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  2/05/07
PHH                           Select Program to Audit            Time: 14:24:18

Type choices, press F10.

Input Source Memb        RTPA Maintenance Menu
  File Name . . .
  Library Name. .    Enter option#, press enter.

Object to Library     1. User Profile Maintenance
Create As . . . .     2. User Standard Audit Options Maintenance
                      3. RPGIII Operation Code Maintenance
Audit File Outq .     4. RPGIV Operation Code Maintenance
Max. Audit Pages      5. Standard Subroutines to be bypassed for Auditing
                      6. Create User RTPA Testing Library (first in *Libl)
                      7. WRKSPLF
Audit Compile Lis     8. Delete Spooled Files for Current User (Sign On)
(Only)                9. WRKSBMJOB *JOB

                    Option#
                               (Clear RTPA Expanded Objects in Lib Z$AUDITE
F1=Help         F    F3=Exit    for All Users with CALL Z$CLRFIL)
F18=Job History
```

Figure 4.23 RTPA Maintenance Menu Options and features

RTPA Maintenance Menu Options and Features

1. User Profile Maintenance

RTPA User Profile Maintenance allows the RTPA User to customize his or her RTPA dynamically created compile override defaults, which are stored in file Z$FI01, and is illustrated in this chapter.

2. User Standard Audit Options Maintenance

RTPA User Profile Maintenance allows the RTPA User to customize his or her RTPA dynamically created compile override defaults, which are stored in file Z$FI02, and is illustrated in chapter 7.

3. RPGIII Operation Code Maintenance (not changed by the User Company)

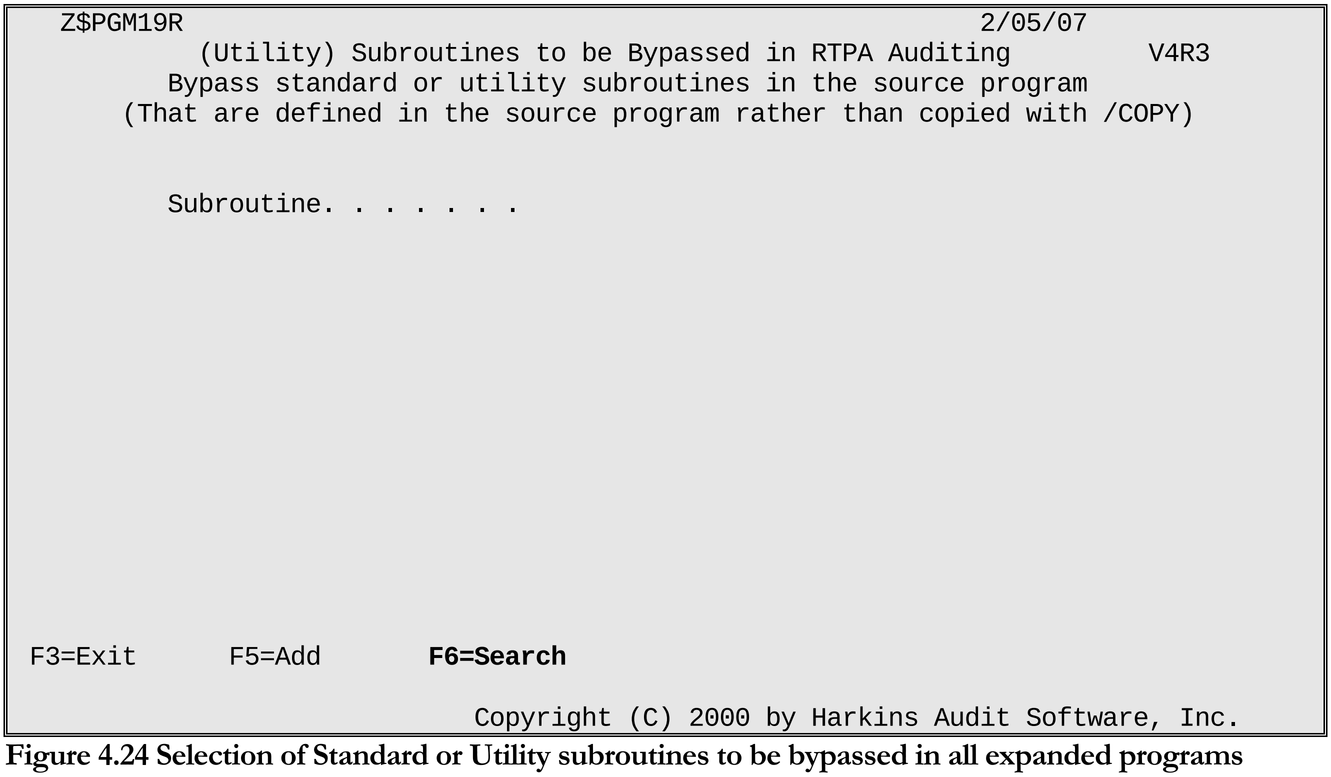
RTPA RPGIII Operation code maintenance allows the addition of more RPGIII RPG Operation codes for auditing by RTPA.
This option is not normally used by the User Company.

4. RPGIV Operation Code Maintenance (not changed by the User Company)

RTPA RPGIV Operation code maintenance allows the addition of more (new) RPGIV RPG Operation codes for auditing by RTPA.
This option is not normally used by the User Company.

5. Standard Utility Subroutines to be bypassed for Auditing

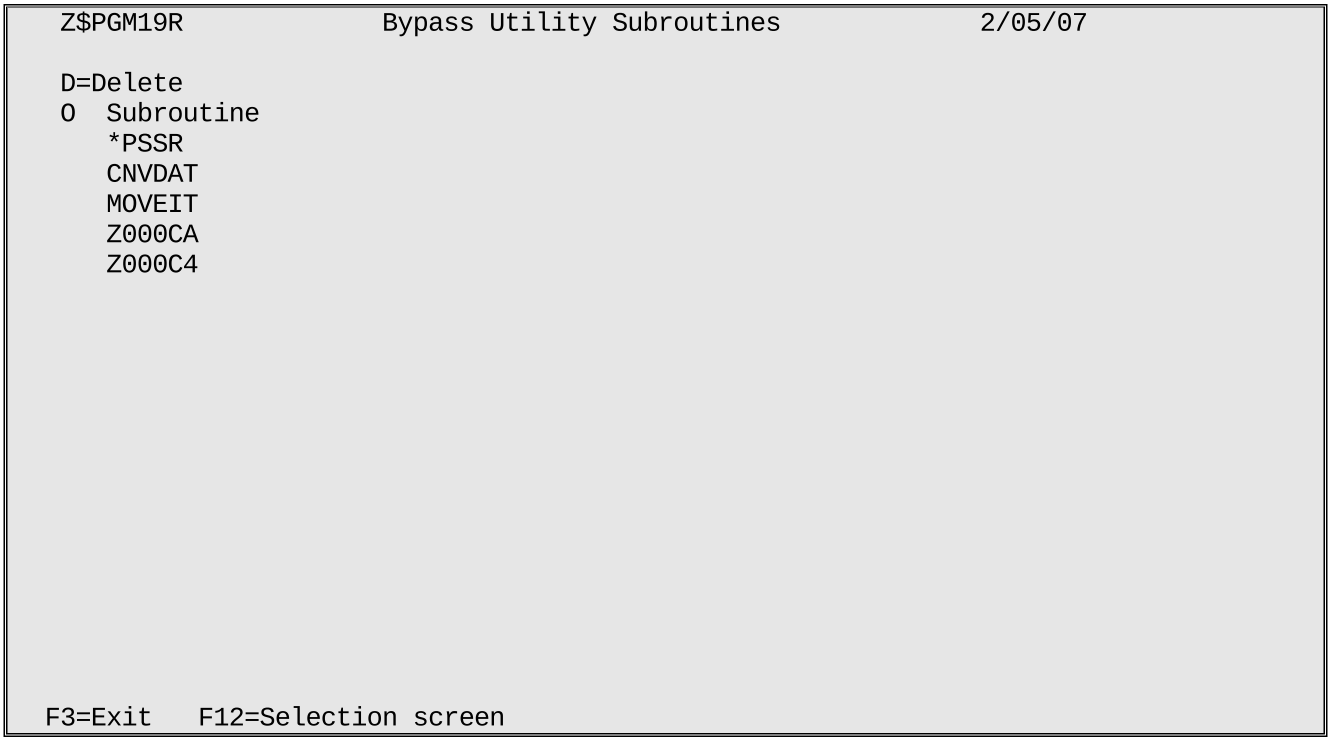
Standard or Utility subroutines (BEGSR) that a Company uses in many programs for standard functions such as retrieving a company name, may be entered in this option.
These subroutines are bypassed for RTPA auditing when the source program using them is expanded by RTPA, saving audit output for proven routines.



```
   Z$PGM19R                                                   2/05/07
            (Utility) Subroutines to be Bypassed in RTPA Auditing        V4R3
          Bypass standard or utility subroutines in the source program
       (That are defined in the source program rather than copied with /COPY)


          Subroutine. . . . . . .















 F3=Exit      F5=Add       F6=Search

                              Copyright (C) 2000 by Harkins Audit Software, Inc.
```

Figure 4.24 Selection of Standard or Utility subroutines to be bypassed in all expanded programs

Command key 6 may be pressed to display the Company standard subroutines to be bypassed for auditing



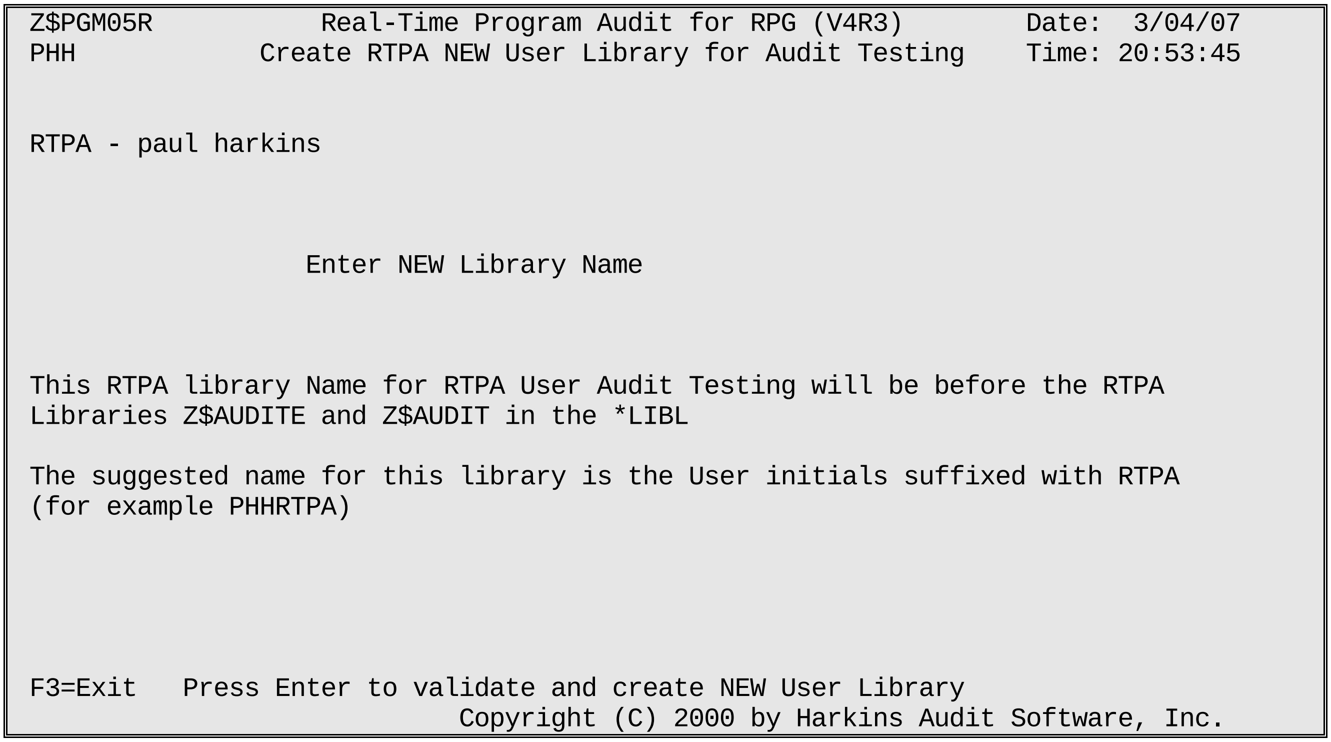
```
   Z$PGM19R             Bypass Utility Subroutines             2/05/07

   D=Delete
   O  Subroutine
      *PSSR
      CNVDAT
      MOVEIT
      Z000CA
      Z000C4










  F3=Exit   F12=Selection screen
```

Figure 4.25 Standard or Utility subroutines to be bypassed in all expanded programs for auditing

6. Create User RTPA Testing Library (first in *Libl)

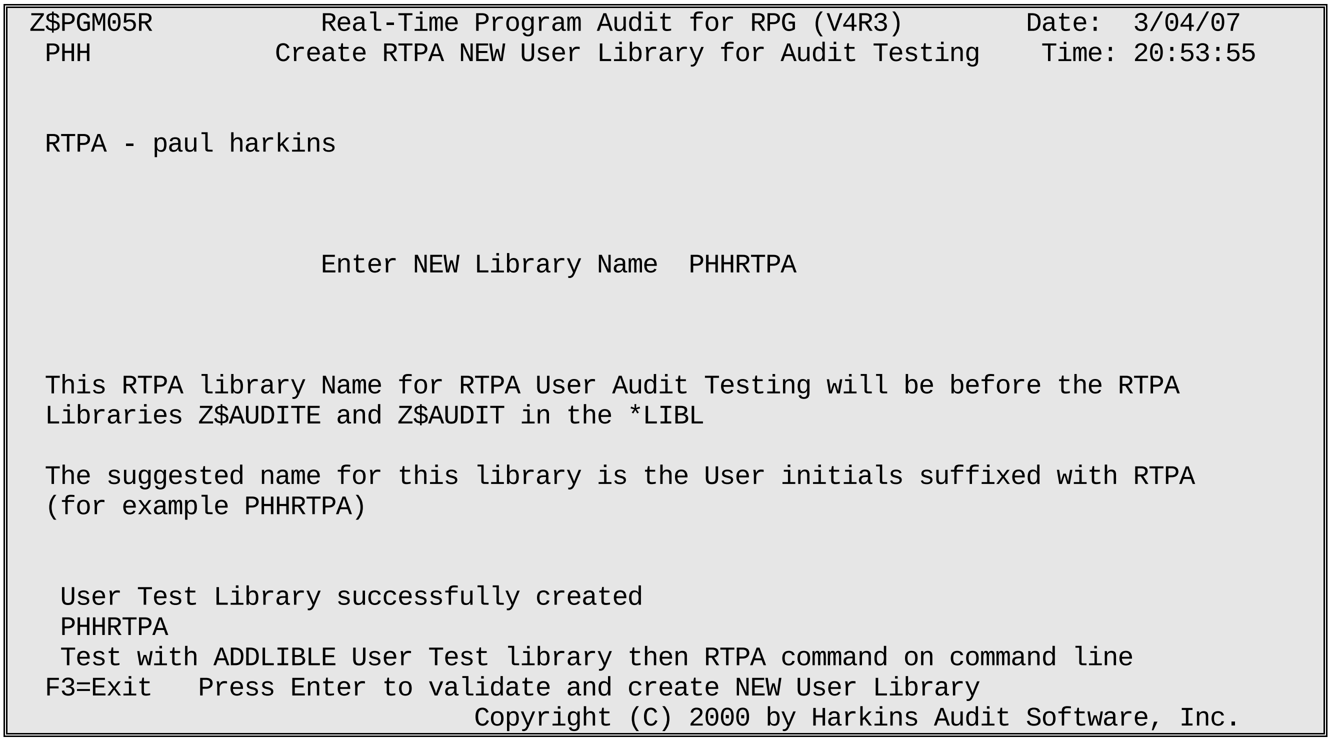
Each RTPA User is encouraged to create a private RTPA auditing library so that RTPA expansions his or her RTPA expanded (audit enabled) object programs may be placed in that library, rather than in the RTPA expanded library Z$AUDITE (which is the default library for all RTPA expanded audit enabled objects.

This allows the resulting RTPA User test library (in this example PHHRTPA) to be placed first in the testing library list for the User, ahead of the RTPA libraries Z$AUDITE, and Z$AUDIT.



```
Z$PGM05R           Real-Time Program Audit for RPG (V4R3)        Date:  3/04/07
PHH            Create RTPA NEW User Library for Audit Testing    Time: 20:53:45


RTPA - paul harkins



                  Enter NEW Library Name



This RTPA library Name for RTPA User Audit Testing will be before the RTPA
Libraries Z$AUDITE and Z$AUDIT in the *LIBL

The suggested name for this library is the User initials suffixed with RTPA
(for example PHHRTPA)






F3=Exit   Press Enter to validate and create NEW User Library
                             Copyright (C) 2000 by Harkins Audit Software, Inc.
```

Figure 4.26 Creation of an RTPA User (private) testing library for RTPA testing



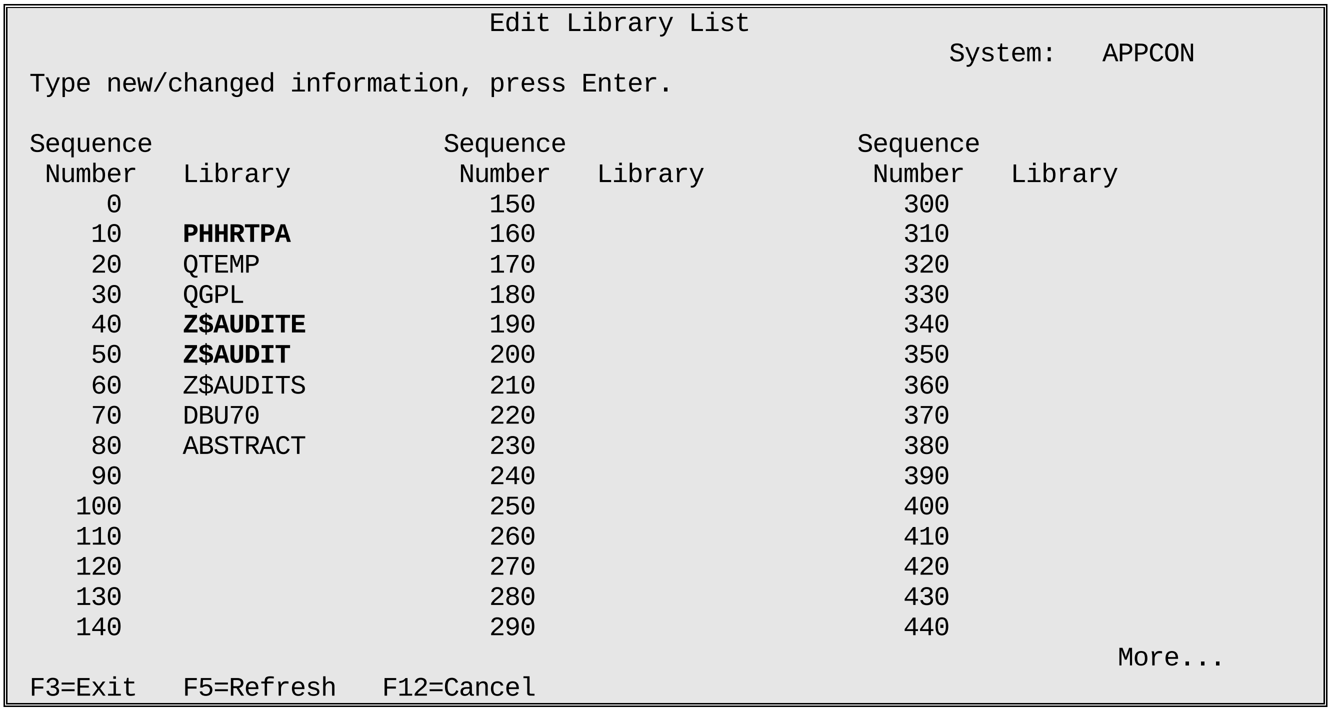
```
 Z$PGM05R           Real-Time Program Audit for RPG (V4R3)       Date:  3/04/07
  PHH            Create RTPA NEW User Library for Audit Testing    Time: 20:53:55


  RTPA - paul harkins



                  Enter NEW Library Name  PHHRTPA



  This RTPA library Name for RTPA User Audit Testing will be before the RTPA
  Libraries Z$AUDITE and Z$AUDIT in the *LIBL

  The suggested name for this library is the User initials suffixed with RTPA
  (for example PHHRTPA)


   User Test Library successfully created
   PHHRTPA
   Test with ADDLIBLE User Test library then RTPA command on command line
  F3=Exit   Press Enter to validate and create NEW User Library
                              Copyright (C) 2000 by Harkins Audit Software, Inc.
```

Figure 4.27 User RTPA Test library PHHRTPA successfully created

User Test Library successfully created

Test with ADDLIBLE User Test library then RTPA command on command line

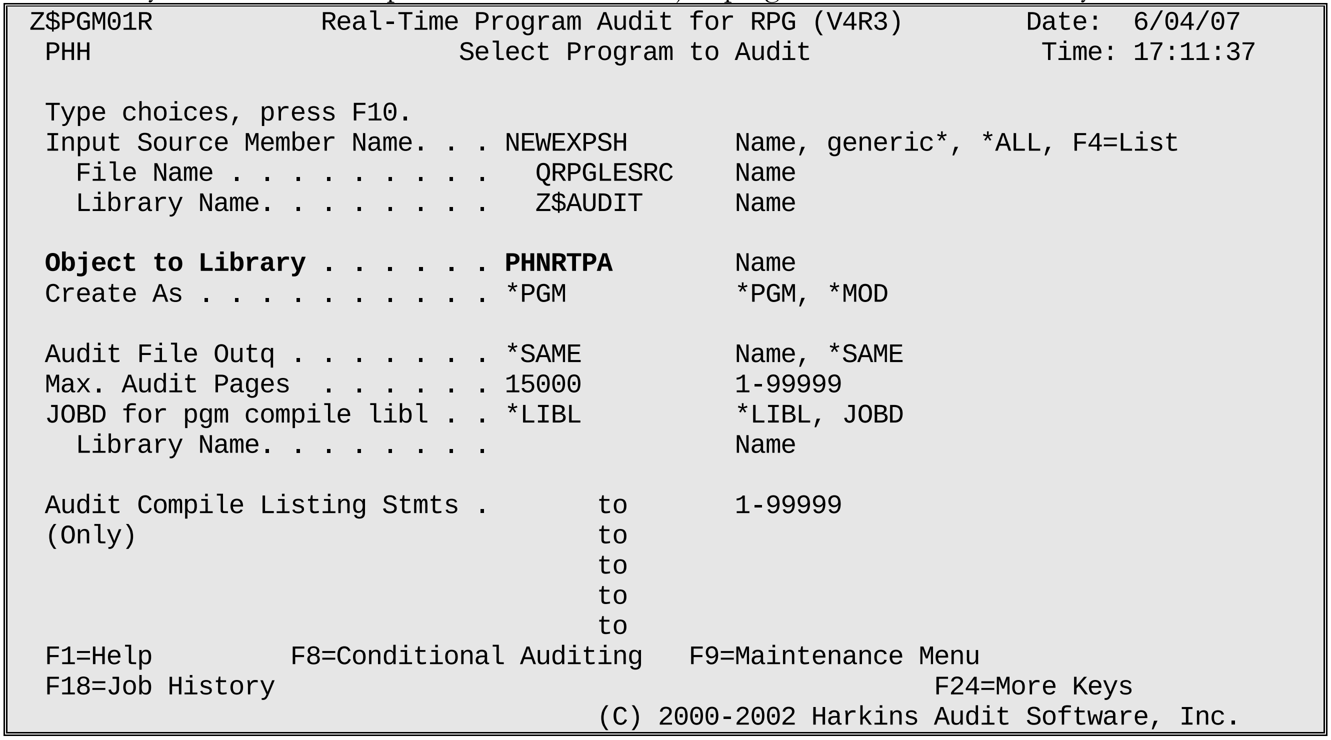
ADDLIBLE PHHRTPA (Your User RTPA Test Library)



```
                              Edit Library List
                                                           System:   APPCON
 Type new/changed information, press Enter.

 Sequence                   Sequence                   Sequence
  Number   Library           Number   Library           Number   Library
      0                        150                        300
     10    PHHRTPA             160                        310
     20    QTEMP               170                        320
     30    QGPL                180                        330
     40    Z$AUDITE            190                        340
     50    Z$AUDIT             200                        350
     60    Z$AUDITS            210                        360
     70    DBU70               220                        370
     80    ABSTRACT            230                        380
     90                        240                        390
    100                        250                        400
    110                        260                        410
    120                        270                        420
    130                        280                        430
    140                        290                        440
                                                                    More...
 F3=Exit   F5=Refresh   F12=Cancel
```

Figure 4.28 User RTPA Test library PHHRTPA is before RTPA libraries Z$AUDITE, Z$AUDIT

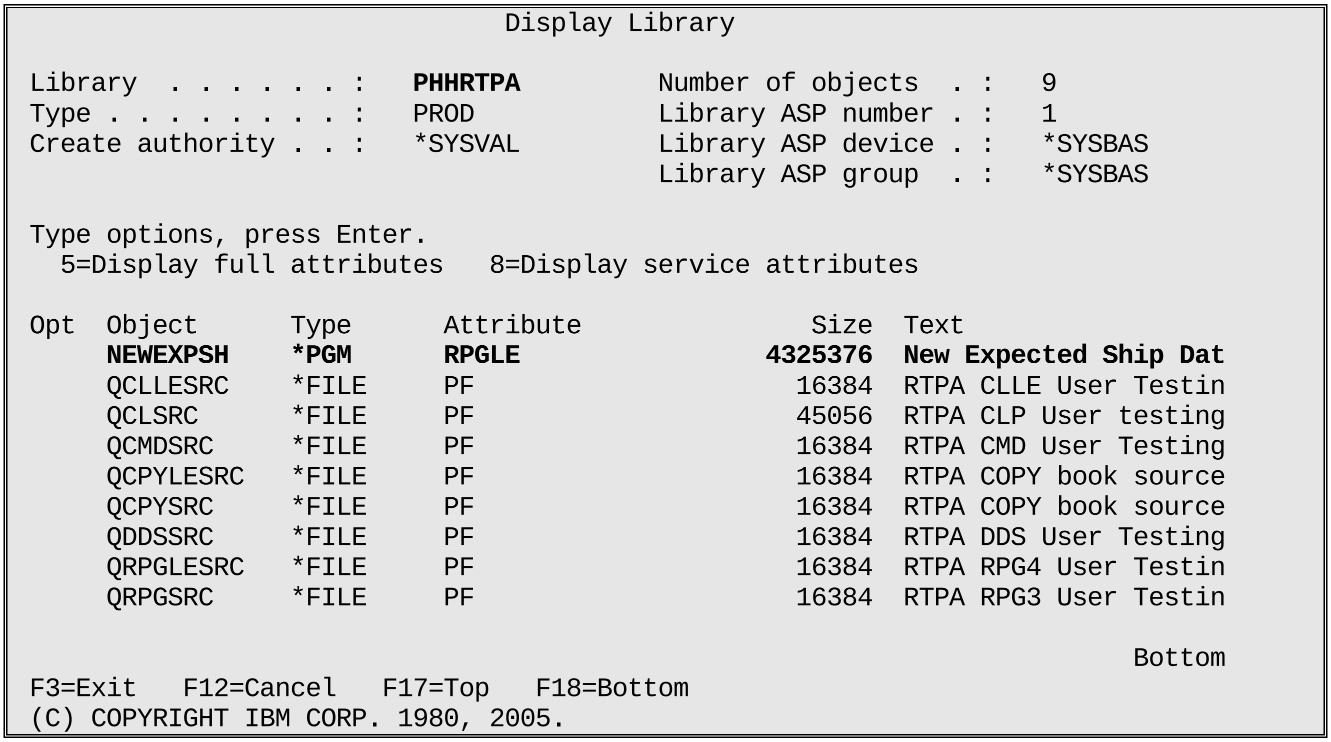
The User may then test RTPA expanded audit enabled object program s from that test library



```
 Z$PGM01R          Real-Time Program Audit for RPG (V4R3)       Date:  6/04/07
  PHH                         Select Program to Audit             Time: 17:11:37

  Type choices, press F10.
  Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
    File Name . . . . . . . . .   QRPGLESRC    Name
    Library Name. . . . . . . .   Z$AUDIT      Name

  Object to Library . . . . . . PHNRTPA        Name
  Create As . . . . . . . . . . *PGM           *PGM, *MOD

  Audit File Outq . . . . . . . *SAME          Name, *SAME
  Max. Audit Pages  . . . . . . 15000          1-99999
  JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
    Library Name. . . . . . . .                Name

  Audit Compile Listing Stmts .       to       1-99999
  (Only)                              to
                                      to
                                      to
                                      to
  F1=Help        F8=Conditional Auditing   F9=Maintenance Menu
  F18=Job History                                         F24=More Keys
                                      (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 4.29 RTPA Expanded object program to User Test library

Program NEWEXPSH placed in library PHHRTPA. 00 highest severity.



```
                                Display Library

Library  . . . . . . :   PHHRTPA         Number of objects  . :   9
Type . . . . . . . . :   PROD            Library ASP number . :   1
Create authority . . :   *SYSVAL         Library ASP device . :   *SYSBAS
                                         Library ASP group  . :   *SYSBAS

Type options, press Enter.
  5=Display full attributes   8=Display service attributes

Opt  Object      Type      Attribute               Size  Text
     NEWEXPSH    *PGM      RPGLE                4325376  New Expected Ship Dat
     QCLLESRC    *FILE     PF                     16384  RTPA CLLE User Testin
     QCLSRC      *FILE     PF                     45056  RTPA CLP User testing
     QCMDSRC     *FILE     PF                     16384  RTPA CMD User Testing
     QCPYLESRC   *FILE     PF                     16384  RTPA COPY book source
     QCPYSRC     *FILE     PF                     16384  RTPA COPY book source
     QDDSSRC     *FILE     PF                     16384  RTPA DDS User Testing
     QRPGLESRC   *FILE     PF                     16384  RTPA RPG4 User Testin
     QRPGSRC     *FILE     PF                     16384  RTPA RPG3 User Testin

                                                                        Bottom
F3=Exit   F12=Cancel   F17=Top   F18=Bottom
(C) COPYRIGHT IBM CORP. 1980, 2005.
```

Figure 4.30 RTPA User Test library

7. WRKSPLF

The WRKSPLF option displays the User Spool File

8. Delete Spooled Files for Current User (Sign On)

The Delete Spool file option deletes all spooled files in the User spool file

9. WRKSBMJOB *JOB

The WRKSBMJOB option displays submitted jobs for the RTPA user (Signon)

(Clear RTPA Expanded Objects in Lib Z$AUDITE for all Users with CALL Z$CLRFIL)

The CALL Z$CLRFIL program is executed from a command line when no RTPA Users are active. This could be executed daily or periodically to cleanup RTPA work files and to clear the RTPA expanded Library Z$AUDITE.
Note – All object programs and modules in the RTPA expanded library Z$AUDITE are deleted when CALL Z$CLRFIL is processed.

RTPA Audit output in Character and Hexadecimal (HEX)

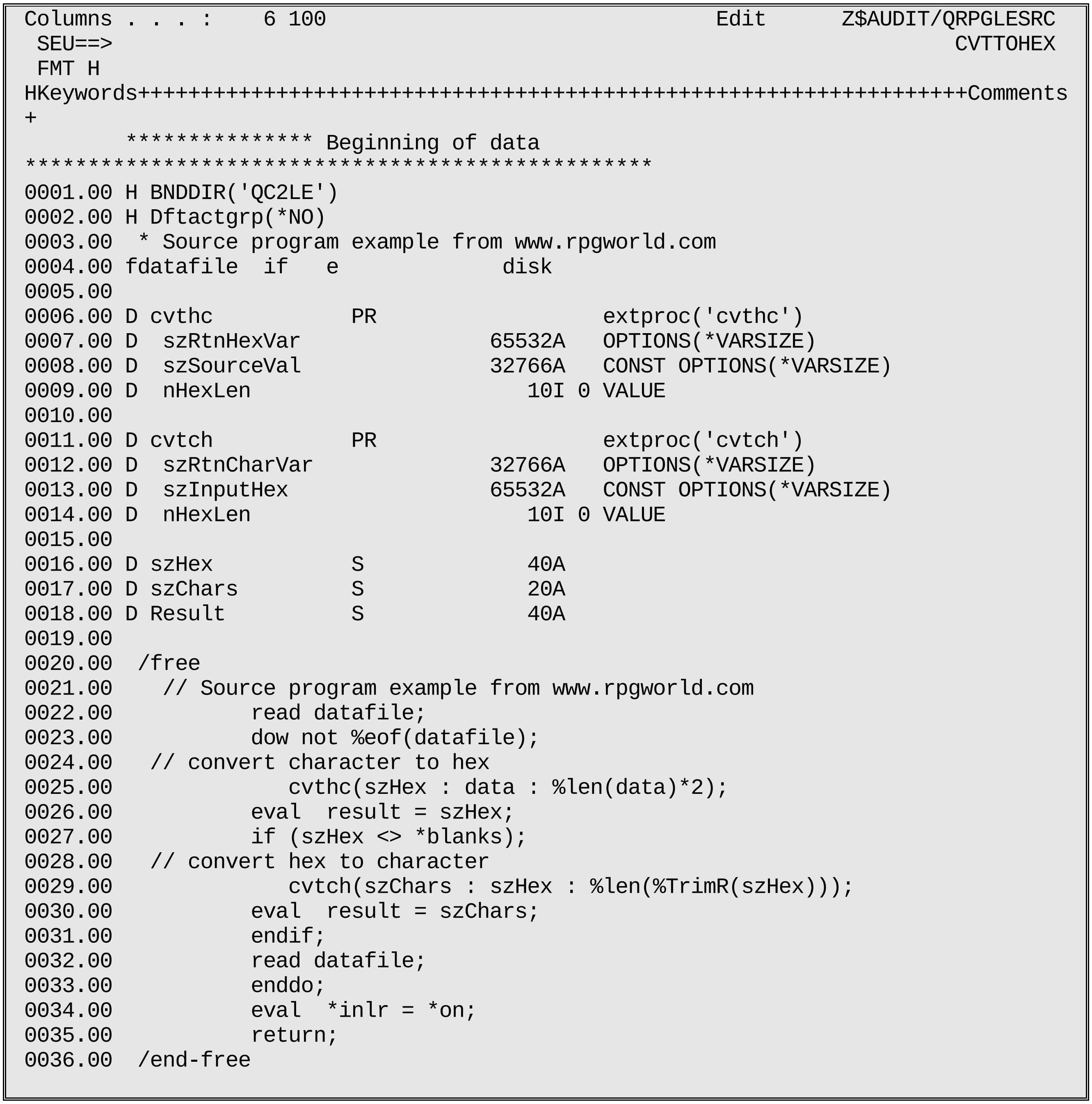
RTPA audit output is to the audit file ZZAUDITP, which is character format for both the source program statements and the data.

RTPA audit output of data may be in both character format and in Hexadecimal format in RTPA Query by selecting the Hexadecimal option.

RTPA converts the audit output data from the normal character format to Hexadecimal by using the conversion code as illustrated in the RPGLE CVTTOHEX sample program, then converts the Hexadecimal to over and under format in the RTPA Query audit output.

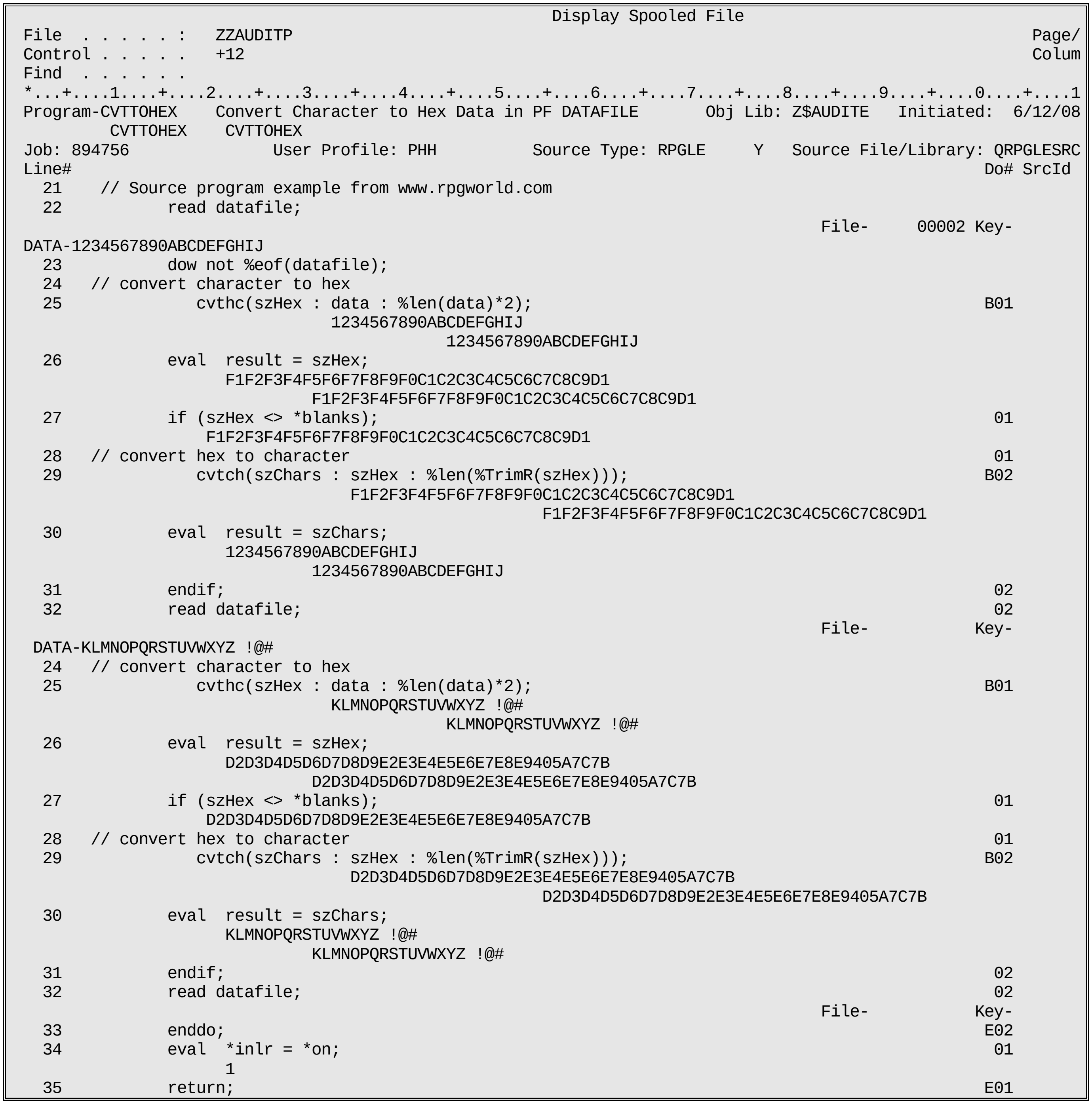
RTPA only show Hexadecimal for special characters, by translating upper case letters, lower case letters, numbers, and the editing characters ,$:.- to blanks before converting data lines to hexadecimal.

Thus, RTPA Query audits data in both character and hexadecimal format where the data is special characters.



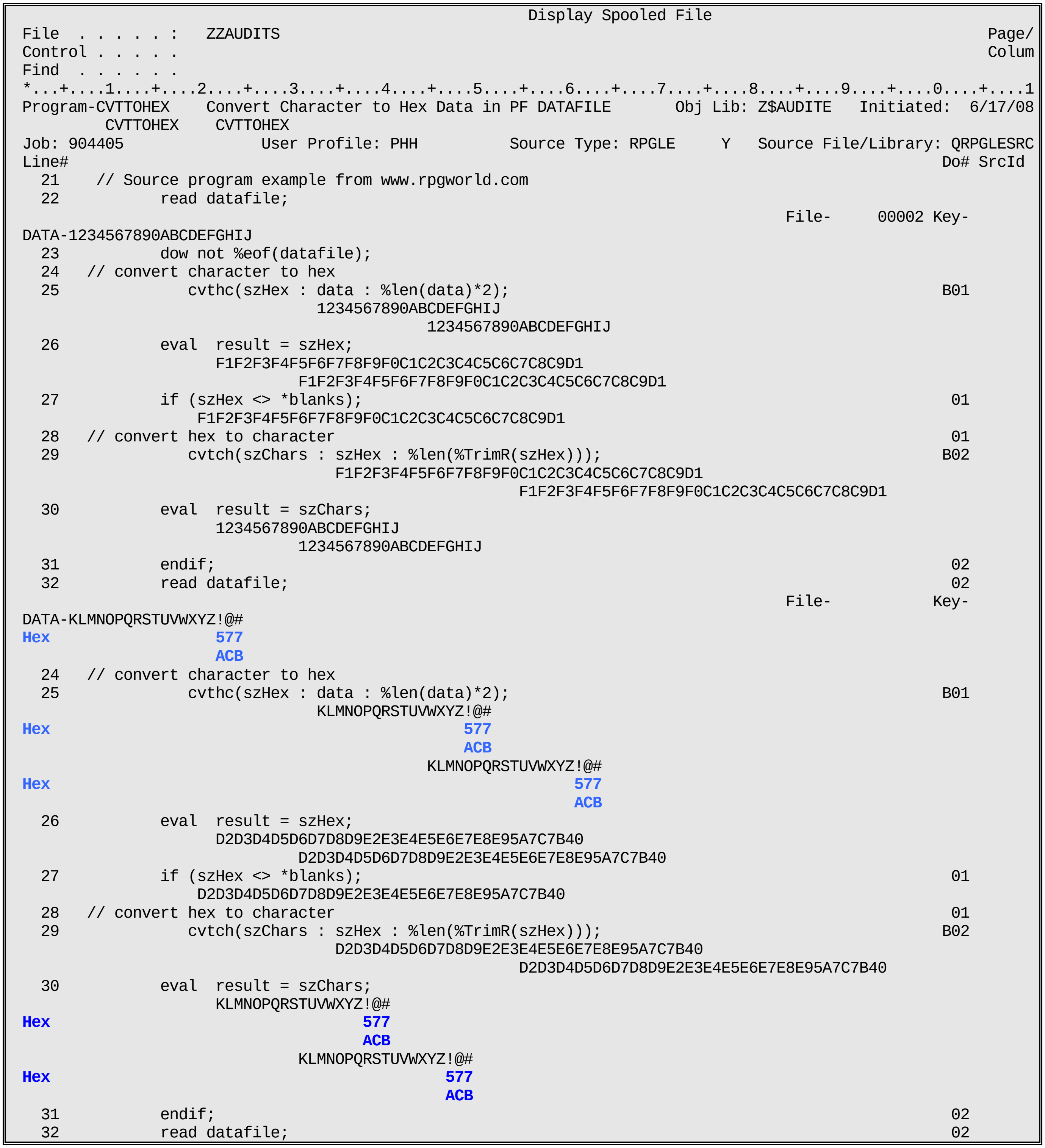
```
Columns . . . :    6 100                               Edit      Z$AUDIT/QRPGLESRC
 SEU==>                                                                   CVTTOHEX
 FMT H
HKeywords++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++Comments
+
        *************** Beginning of data
**************************************************
H BNDDIR('QC2LE')
H Dftactgrp(*NO)
 * Source program example from www.rpgworld.com
fdatafile  if   e             disk

D cvthc           PR                  extproc('cvthc')
D  szRtnHexVar               65532A   OPTIONS(*VARSIZE)
D  szSourceVal               32766A   CONST OPTIONS(*VARSIZE)
D  nHexLen                      10I 0 VALUE

D cvtch           PR                  extproc('cvtch')
D  szRtnCharVar              32766A   OPTIONS(*VARSIZE)
D  szInputHex                65532A   CONST OPTIONS(*VARSIZE)
D  nHexLen                      10I 0 VALUE

D szHex           S             40A
D szChars         S             20A
D Result          S             40A

 /free
   // Source program example from www.rpgworld.com
          read datafile;
          dow not %eof(datafile);
  // convert character to hex
             cvthc(szHex : data : %len(data)*2);
          eval  result = szHex;
          if (szHex <> *blanks);
  // convert hex to character
             cvtch(szChars : szHex : %len(%TrimR(szHex)));
          eval  result = szChars;
          endif;
          read datafile;
          enddo;
          eval  *inlr = *on;
          return;
 /end-free
```

Figure 4.31 CVTTOHEX sample source program



```
                                                       Display Spooled File
File  . . . . . :   ZZAUDITP                                                                             Page/
Control . . . . .   +12                                                                                  Colum
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....0....+....1
Program-CVTTOHEX    Convert Character to Hex Data in PF DATAFILE       Obj Lib: Z$AUDITE   Initiated:  6/12/08
         CVTTOHEX    CVTTOHEX
Job: 894756               User Profile: PHH          Source Type: RPGLE     Y   Source File/Library: QRPGLESRC
Line#                                                                                               Do# SrcId
  21    // Source program example from www.rpgworld.com
  22           read datafile;
                                                                                   File-     00002 Key-
DATA-1234567890ABCDEFGHIJ
  23           dow not %eof(datafile);
  24   // convert character to hex
  25              cvthc(szHex : data : %len(data)*2);                                               B01
                                1234567890ABCDEFGHIJ
                                            1234567890ABCDEFGHIJ
  26           eval  result = szHex;
                     F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
                              F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
  27           if (szHex <> *blanks);                                                                01
                   F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
  28   // convert hex to character                                                                   01
  29              cvtch(szChars : szHex : %len(%TrimR(szHex)));                                     B02
                                  F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
                                                      F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
  30           eval  result = szChars;
                     1234567890ABCDEFGHIJ
                              1234567890ABCDEFGHIJ
  31           endif;                                                                                02
  32           read datafile;                                                                        02
                                                                                   File-           Key-
 DATA-KLMNOPQRSTUVWXYZ !@#
  24   // convert character to hex
  25              cvthc(szHex : data : %len(data)*2);                                               B01
                                KLMNOPQRSTUVWXYZ !@#
                                            KLMNOPQRSTUVWXYZ !@#
  26           eval  result = szHex;
                     D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9405A7C7B
                              D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9405A7C7B
  27           if (szHex <> *blanks);                                                                01
                   D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9405A7C7B
  28   // convert hex to character                                                                   01
  29              cvtch(szChars : szHex : %len(%TrimR(szHex)));                                     B02
                                  D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9405A7C7B
                                                      D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E9405A7C7B
  30           eval  result = szChars;
                     KLMNOPQRSTUVWXYZ !@#
                              KLMNOPQRSTUVWXYZ !@#
  31           endif;                                                                                02
  32           read datafile;                                                                        02
                                                                                   File-           Key-
  33           enddo;                                                                               E02
  34           eval  *inlr = *on;                                                                    01
                     1
  35           return;                                                                              E01
```

Figure 4.32 CVTTOHEX sample source program RTPA audit output



```
                                                       Display Spooled File
File  . . . . . :   ZZAUDITS                                                                             Page/
Control . . . . .                                                                                        Colum
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+....8....+....9....+....0....+....1
Program-CVTTOHEX    Convert Character to Hex Data in PF DATAFILE       Obj Lib: Z$AUDITE   Initiated:  6/17/08
         CVTTOHEX    CVTTOHEX
Job: 904405               User Profile: PHH          Source Type: RPGLE     Y   Source File/Library: QRPGLESRC
Line#                                                                                               Do# SrcId
  21    // Source program example from www.rpgworld.com
  22           read datafile;
                                                                                   File-     00002 Key-
DATA-1234567890ABCDEFGHIJ
  23           dow not %eof(datafile);
  24   // convert character to hex
  25              cvthc(szHex : data : %len(data)*2);                                               B01
                                1234567890ABCDEFGHIJ
                                            1234567890ABCDEFGHIJ
  26           eval  result = szHex;
                     F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
                              F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
  27           if (szHex <> *blanks);                                                                01
                   F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
  28   // convert hex to character                                                                   01
  29              cvtch(szChars : szHex : %len(%TrimR(szHex)));                                     B02
                                  F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
                                                      F1F2F3F4F5F6F7F8F9F0C1C2C3C4C5C6C7C8C9D1
  30           eval  result = szChars;
                     1234567890ABCDEFGHIJ
                              1234567890ABCDEFGHIJ
  31           endif;                                                                                02
  32           read datafile;                                                                        02
                                                                                   File-           Key-
DATA-KLMNOPQRSTUVWXYZ!@#
Hex                  577
                     ACB
  24   // convert character to hex
  25              cvthc(szHex : data : %len(data)*2);                                               B01
                                KLMNOPQRSTUVWXYZ!@#
Hex                                             577
                                                ACB
                                            KLMNOPQRSTUVWXYZ!@#
Hex                                                         577
                                                            ACB
  26           eval  result = szHex;
                     D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E95A7C7B40
                              D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E95A7C7B40
  27           if (szHex <> *blanks);                                                                01
                   D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E95A7C7B40
  28   // convert hex to character                                                                   01
  29              cvtch(szChars : szHex : %len(%TrimR(szHex)));                                     B02
                                  D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E95A7C7B40
                                                      D2D3D4D5D6D7D8D9E2E3E4E5E6E7E8E95A7C7B40
  30           eval  result = szChars;
                     KLMNOPQRSTUVWXYZ!@#
Hex                                  577
                                     ACB
                              KLMNOPQRSTUVWXYZ!@#
Hex                                          577
                                             ACB
  31           endif;                                                                                02
  32           read datafile;                                                                        02
```

Figure 4.33 RTPA Query audit output of CVTTOHEX program showing Hex over and under

Chapter

5

Chapter 5: **Advanced Auditing (Focused Auditing)**

RTPA offers powerful options for controlling exactly what information is to be audited with its Advanced or focused Auditing feature.

Advanced or focused auditing provides the User with very powerful selective auditing options, including selecting auditing of only statements with specific fields (variables), or specific select operation codes..

To access these options, RTPA must analyze the source program to gather some information about it by first compiling the input source program when the programmer presses the F11 key. Once RTPA has completed its analysis of the input source program from the compile listing, the programmer may customize the audit expansion by using the command keys at the bottom of the RTPA selection screen, such as selecting specific variables to audit with the F16 key.

After all the advanced auditing options are selected, the programmer presses the F10 to submit the program for expansion with the selected audits. Audited programs are almost always completely audited, with virtually all the executing source statements being audited with the F10 submit program, however the F11 advanced or focused auditing feature can provide focused audits of areas of interest to the programmer or auditor.

- ❖ To use the customize function, **press F11** from the main RTPA screen to first compile the input source program and allow RTPA for RPG to build files with the variables, files, operations, subroutines, etc. that are used in the input source program.

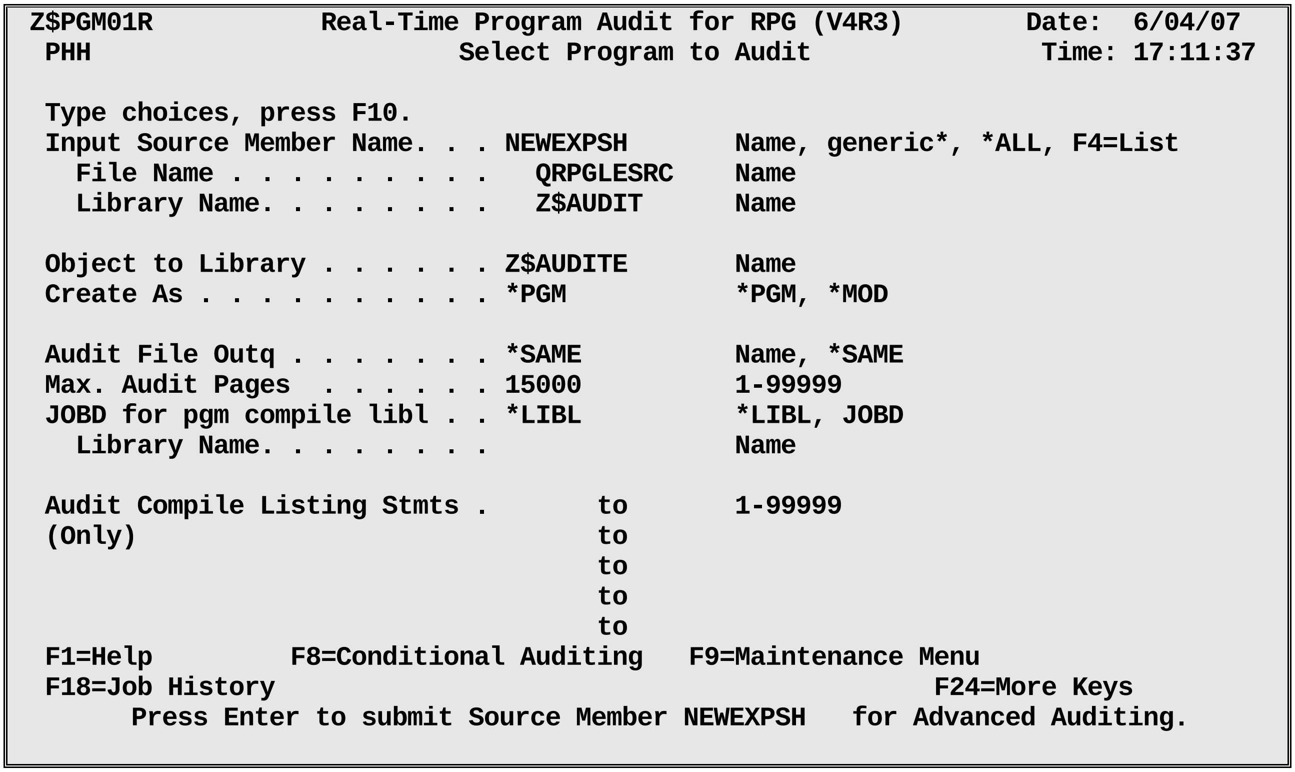
Advanced Auditing gives access to a high degree of control over what information can be included and excluded from an audit. The advanced auditing options include:

- ❖ Select Operations to Audit (F15)
- ❖ Select Labels and Subroutines To Audit (F16)
- ❖ Select Indicators to Audit (F17)
- ❖ Display Called Programs (F20)
- ❖ Select Files to Audit (F22)

Using the F11 Command Key to compile the input source

After the program name, source file and source library are entered, press F11 **(F11=Advanced Auditing),** rather than the F10 key to submit the input source program for a compile, and to build the necessary RTPA work files needed for advanced auditing.

After the F11 key has been pressed to submit the input source program for a compile, the following message will appear:

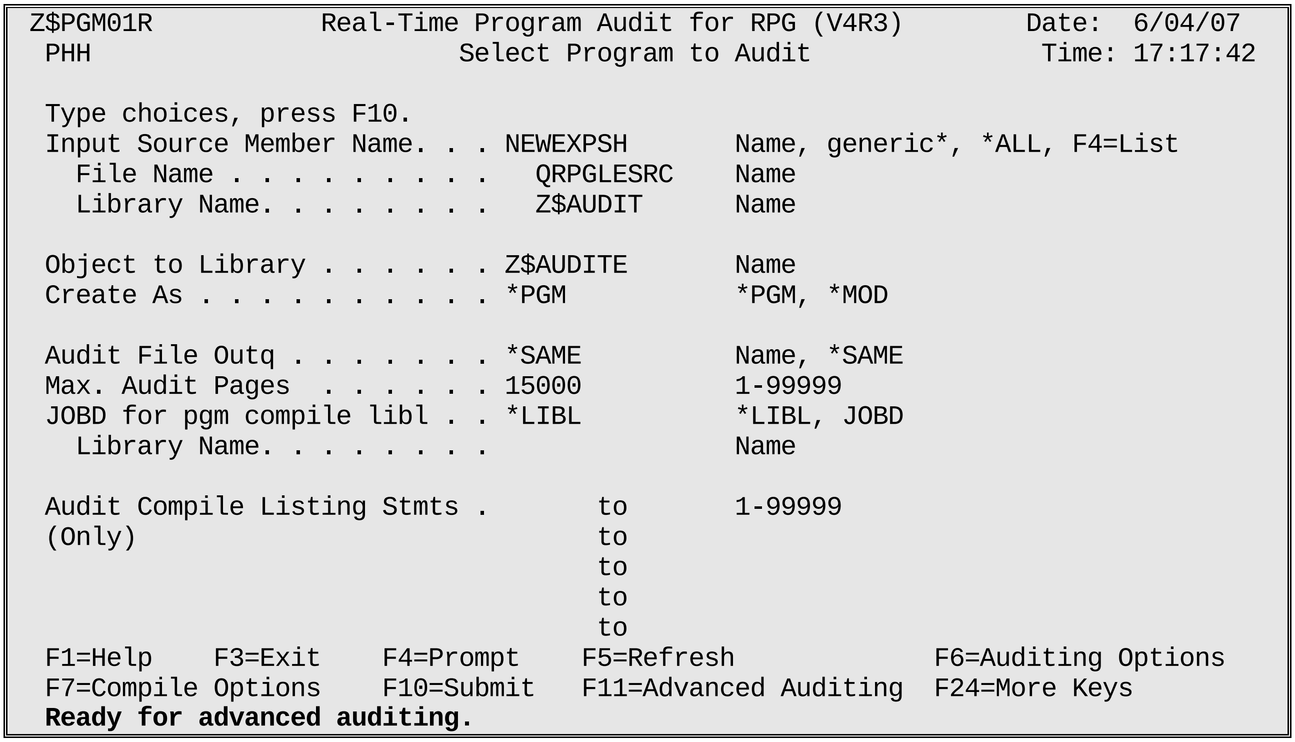


```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)       Date:  6/04/07
  PHH                        Select Program to Audit               Time: 17:11:37

  Type choices, press F10.
  Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
    File Name . . . . . . . . .   QRPGLESRC    Name
    Library Name. . . . . . . .   Z$AUDIT      Name

  Object to Library . . . . . . Z$AUDITE       Name
  Create As . . . . . . . . . . *PGM           *PGM, *MOD

  Audit File Outq . . . . . . . *SAME          Name, *SAME
  Max. Audit Pages  . . . . . . 15000          1-99999
  JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
    Library Name. . . . . . . .                Name

  Audit Compile Listing Stmts .       to       1-99999
  (Only)                              to
                                      to
                                      to
                                      to
  F1=Help         F8=Conditional Auditing   F9=Maintenance Menu
  F18=Job History                                          F24=More Keys
       Press Enter to submit Source Member NEWEXPSH   for Advanced Auditing.
```

Figure 5.1 Program NEWEXPSH selected for Advanced Auditing with Command key 11 (this will submit the input source program for a compile listing)
Press Enter to submit Source Member NEWEXPSH for Advanced Auditing.

The necessary RTPA work files will be created, and the program is ready for selection of the advanced auditing when the following message appears:

```
Ready for advanced auditing.
```



```
Z$PGM01R          Real-Time Program Audit for RPG (V4R3)        Date:  6/04/07
 PHH                         Select Program to Audit              Time: 17:17:42

 Type choices, press F10.
 Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QRPGLESRC    Name
   Library Name. . . . . . . .   Z$AUDIT      Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 15000          1-99999
 JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
   Library Name. . . . . . . .                Name

 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F1=Help    F3=Exit    F4=Prompt    F5=Refresh             F6=Auditing Options
 F7=Compile Options    F10=Submit   F11=Advanced Auditing  F24=More Keys
 Ready for advanced auditing.
```

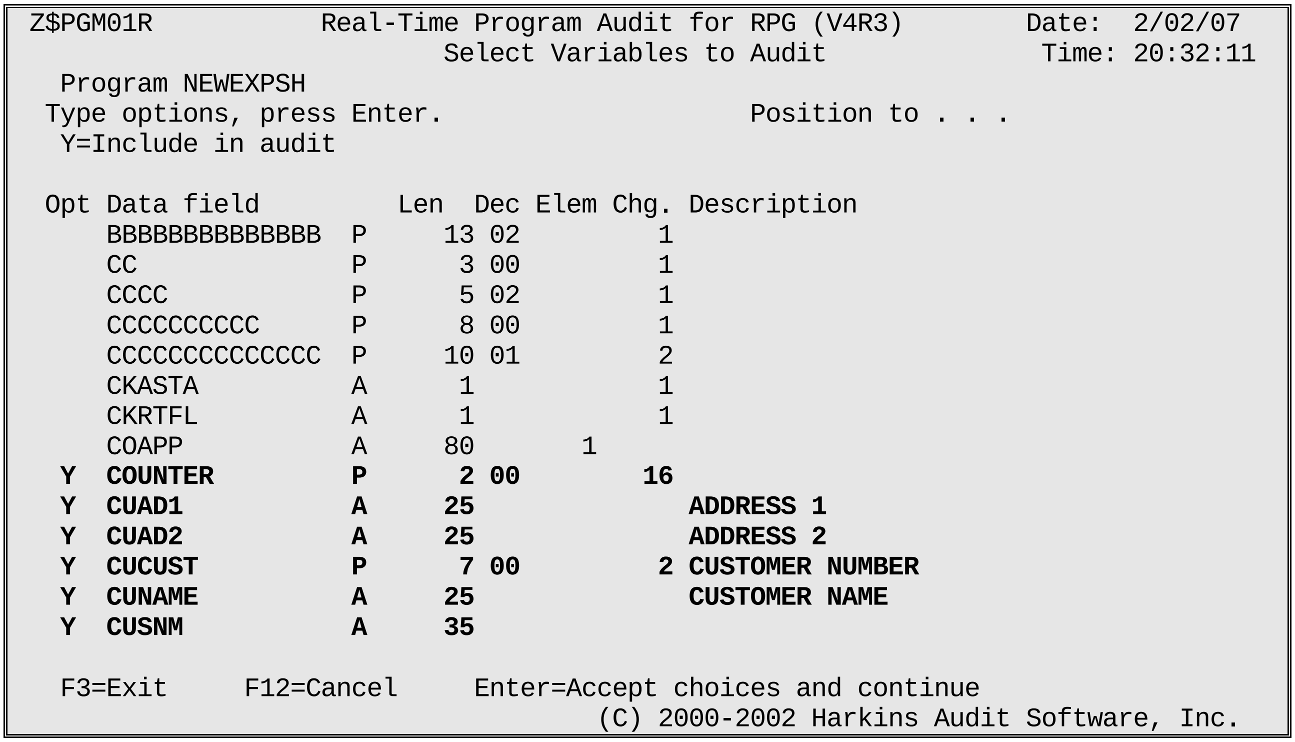
Figure 5.2 Program NEWEXPSH is ready for selection of audit information

The programmer may now use the command keys to selectively audit the program variables, files, subroutines, operation codes, subroutines, etc., confirming each selection with F5 to accept.

Using the F16 Command Key to audit desired variables

In addition, only executing source statements with the selected variables (or Operation codes) may be optionally audited.

In the following screen, the F16 command key was pressed to select some variables for auditing. Only the variables selected with Y will have their data values shown in the audit output.

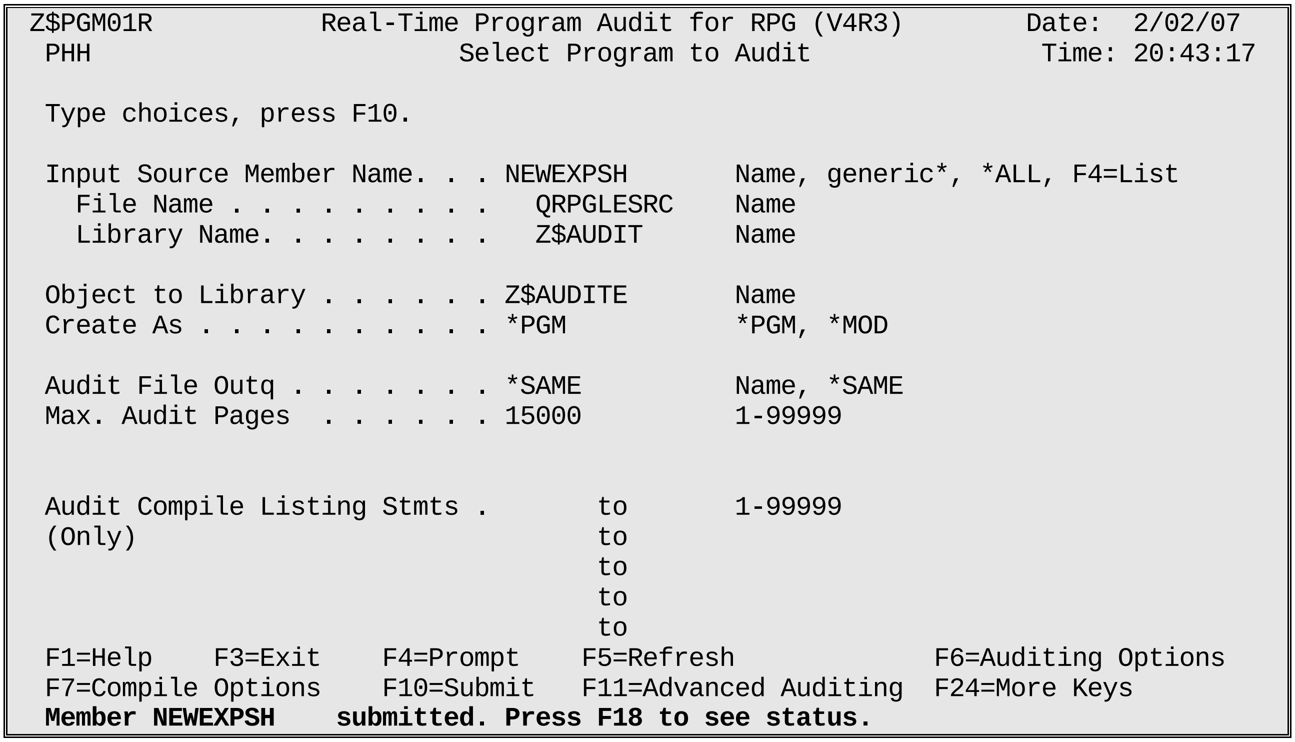


```
 Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  2/02/07
                            Select Variables to Audit              Time: 20:32:11
   Program NEWEXPSH
  Type options, press Enter.                        Position to . . .
   Y=Include in audit

  Opt Data field         Len  Dec Elem Chg. Description
      BBBBBBBBBBBBBB  P     13 02         1
      CC              P      3 00         1
      CCCC            P      5 02         1
      CCCCCCCCCC      P      8 00         1
      CCCCCCCCCCCCCC  P     10 01         2
      CKASTA          A      1            1
      CKRTFL          A      1            1
      COAPP           A     80       1
   Y  COUNTER         P      2 00        16
   Y  CUAD1           A     25              ADDRESS 1
   Y  CUAD2           A     25              ADDRESS 2
   Y  CUCUST          P      7 00         2 CUSTOMER NUMBER
   Y  CUNAME          A     25              CUSTOMER NAME
   Y  CUSNM           A     35

   F3=Exit     F12=Cancel     Enter=Accept choices and continue
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 5.3 Selection of variables to be audited with command key 16 in advanced auditing

Only the program variables with a Y to the left of the variable name will be audited with their contents.

After all advanced or focused auditing selections are made, the F10 command key should be pressed to submit the program for expansion, as the F10 command key is used in expanding the entire program for auditing.



```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)       Date:  2/02/07
 PHH                          Select Program to Audit              Time: 20:43:17

 Type choices, press F10.

 Input Source Member Name. . . NEWEXPSH       Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QRPGLESRC    Name
   Library Name. . . . . . . .   Z$AUDIT      Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 15000          1-99999


 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F1=Help    F3=Exit    F4=Prompt    F5=Refresh             F6=Auditing Options
 F7=Compile Options    F10=Submit   F11=Advanced Auditing  F24=More Keys
 Member NEWEXPSH    submitted. Press F18 to see status.
```

Figure 5.4 Program NEWEXPSH submitted for expansion in advanced auditing

Chapter

6

Chapter 6: **Working with Audit Files**

When you execute an audit-enabled program, the program creates an audit file as part of its own operations. This chapter explains how to use these files.

Reading Audit Files (WRKSPLF and PDF files)

Audit files are sent to the printer queue ZZAUDITP. The audit files are identified the name of the executed program in the data field.

- ❖ Use the IBM command WRKSPLF to display the audit file. Type:

```
WRKSPLF
```

- ❖ Go to the bottom of the spool file listing to get the most recent files. Tab next to the file that you want to view and select **option 5** to display the compile listing. S**elect option P** to display the compile listing as a searchable PDF, if the appropriate IBM programs are available.

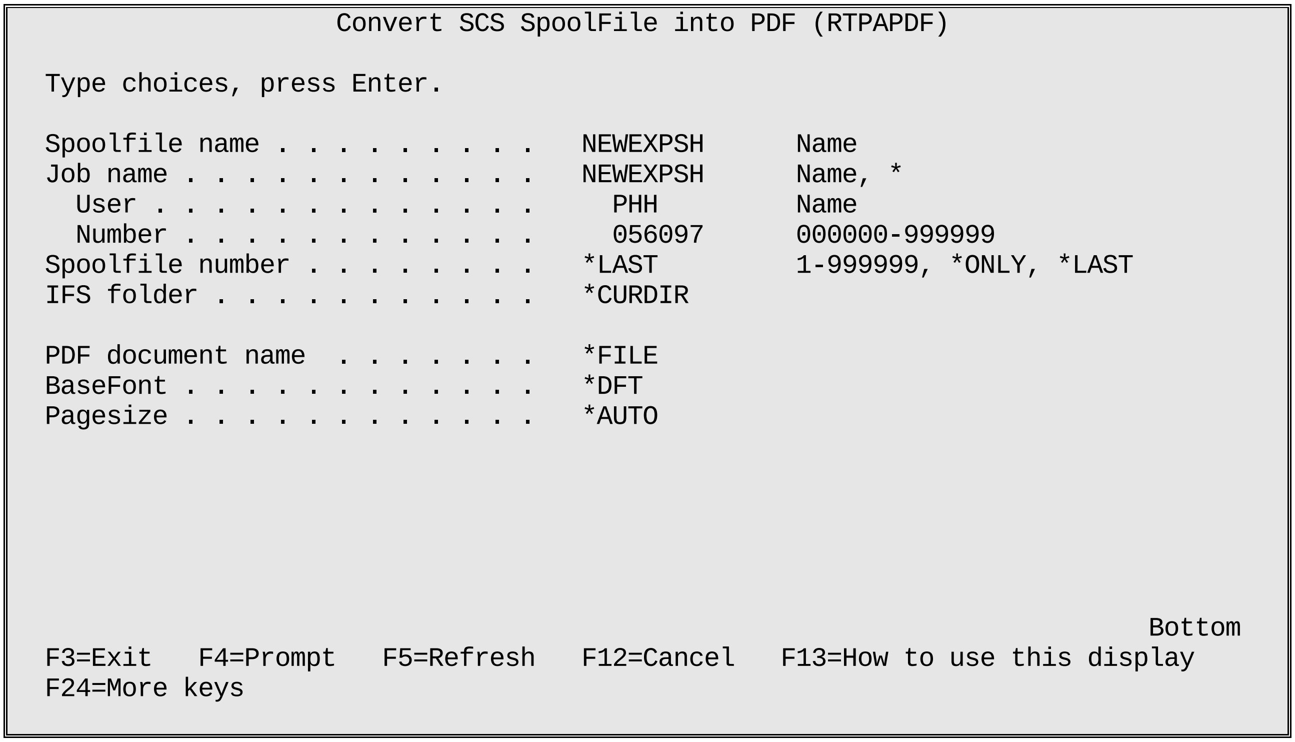
After opening the audit file with WRKSPLF, you can use the Display Spooled File's powerful FIND capability to scan the audit file by:

- Any of the fields in the audited source statement
- The data contents of variables in the audited source statement
- The exact time the source statement was executed
- The input source program compile listing statement sequence number

Converting spool files to PDF files on the IFS

The RTPAPDF command converts a spool file into a searchable PDF file on the IFS

Note – The IBM iSeries Access for Web and IBM Infoprint products may be needed to create and fully search PDF files on the IFS.
The RTPA command RTPAPDF will convert a WRKSPLF spool file into a searchable PDF file on the IFS (Integrated File System), as illustrated below. This allows the compile listing in the IFS file to be fully searched using the PDF search facility



```
                     Convert SCS SpoolFile into PDF (RTPAPDF)

 Type choices, press Enter.

 Spoolfile name . . . . . . . . .   NEWEXPSH      Name
 Job name . . . . . . . . . . . .   NEWEXPSH      Name, *
   User . . . . . . . . . . . . .     PHH         Name
   Number . . . . . . . . . . . .     056097      000000-999999
 Spoolfile number . . . . . . . .   *LAST         1-999999, *ONLY, *LAST
 IFS folder . . . . . . . . . . .   *CURDIR

 PDF document name  . . . . . . .   *FILE
 BaseFont . . . . . . . . . . . .   *DFT
 Pagesize . . . . . . . . . . . .   *AUTO







                                                                          Bottom
 F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
 F24=More keys
```

Figure 6.1 Converting a WRKSPLF RTPA NEWEXPSH audit file to a searchable PDF file in the IFS

Spool file NEWEXPSH converted into /NEWEXPSH.pdf.   (in the System i IFS)

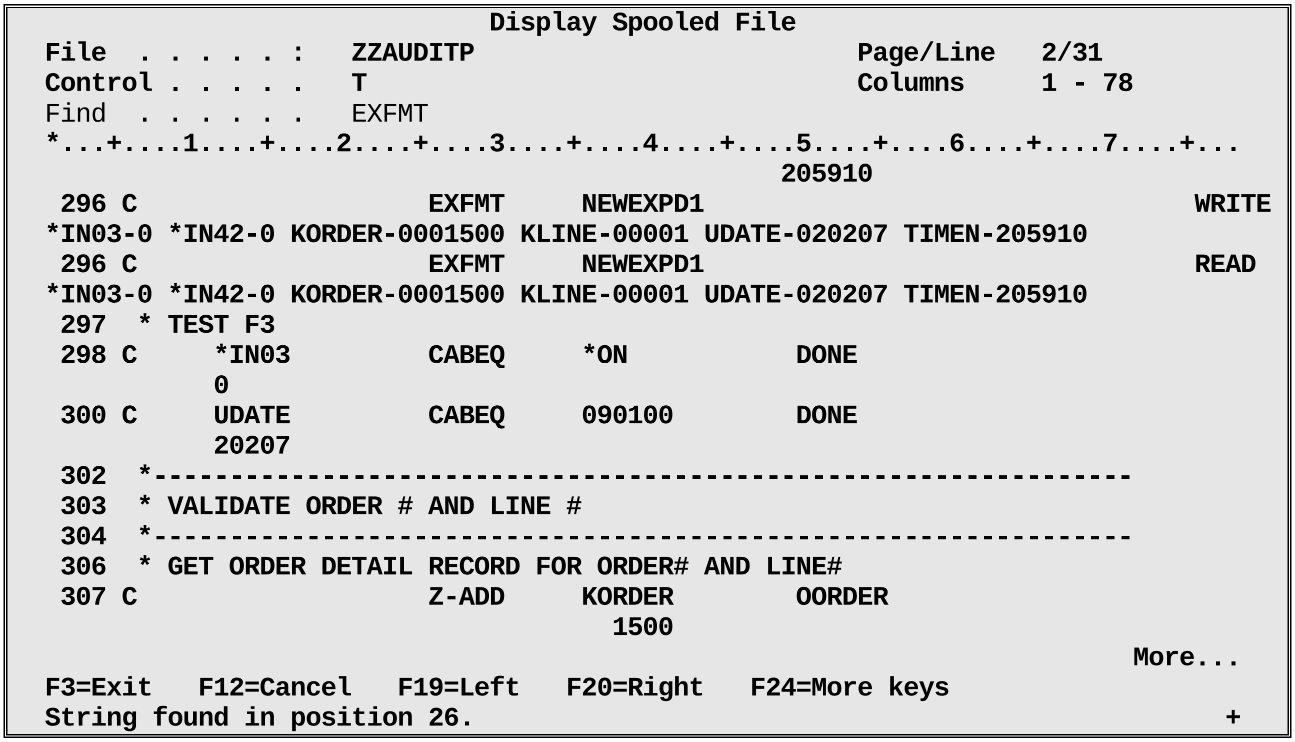
In RPGIV, the audit file can also include:

- The source statement sequence number
- The source statement change ID (positions one through five of the source statement)
- The source statement change date, and a date compare code
- The exact time the instruction was executed to the millisecond
- The Do # level (eg. B01, E01)

TIP: To update the display in real-time while the audit-enabled object is executing, **press systems attention**, **enter 3** (to review the program status), and then **enter 4** to display the spooled files.

Searching the ZZAUDITP Audit file with the FIND feature

The Display Spool File Utility Find capability (F16) will search through the Spool file output for the desired character string.



```
                             Display Spooled File
File  . . . . . :   ZZAUDITP                         Page/Line   2/31
Control . . . . .   T                                Columns     1 - 78
Find  . . . . . .   EXFMT
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
                                                 205910
 296 C                  EXFMT     NEWEXPD1                       WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-020207 TIMEN-205910
 296 C                  EXFMT     NEWEXPD1                       READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-020207 TIMEN-205910
 297  * TEST F3
 298 C     *IN03        CABEQ     *ON          DONE
          0
 300 C     UDATE        CABEQ     090100       DONE
          20207
 302  *-----------------------------------------------------------
 303  * VALIDATE ORDER # AND LINE #
 304  *-----------------------------------------------------------
 306  * GET ORDER DETAIL RECORD FOR ORDER# AND LINE#
 307 C                  Z-ADD     KORDER       OORDER
                                    1500
                                                                    More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
String found in position 26.                                             +
```

Figure 6.2 Using the Find capability to search and review ZZAUDITP files

The Display Spool File Utility Find capability (F16) will search through the Spool file output for the desired character string.

In this example **Find . . . . EXFMT** and command key 16, will locate the first time the EXFMT Operation code was executed in this execution of the NEWEXPSH program, with this data.

Chapter

7

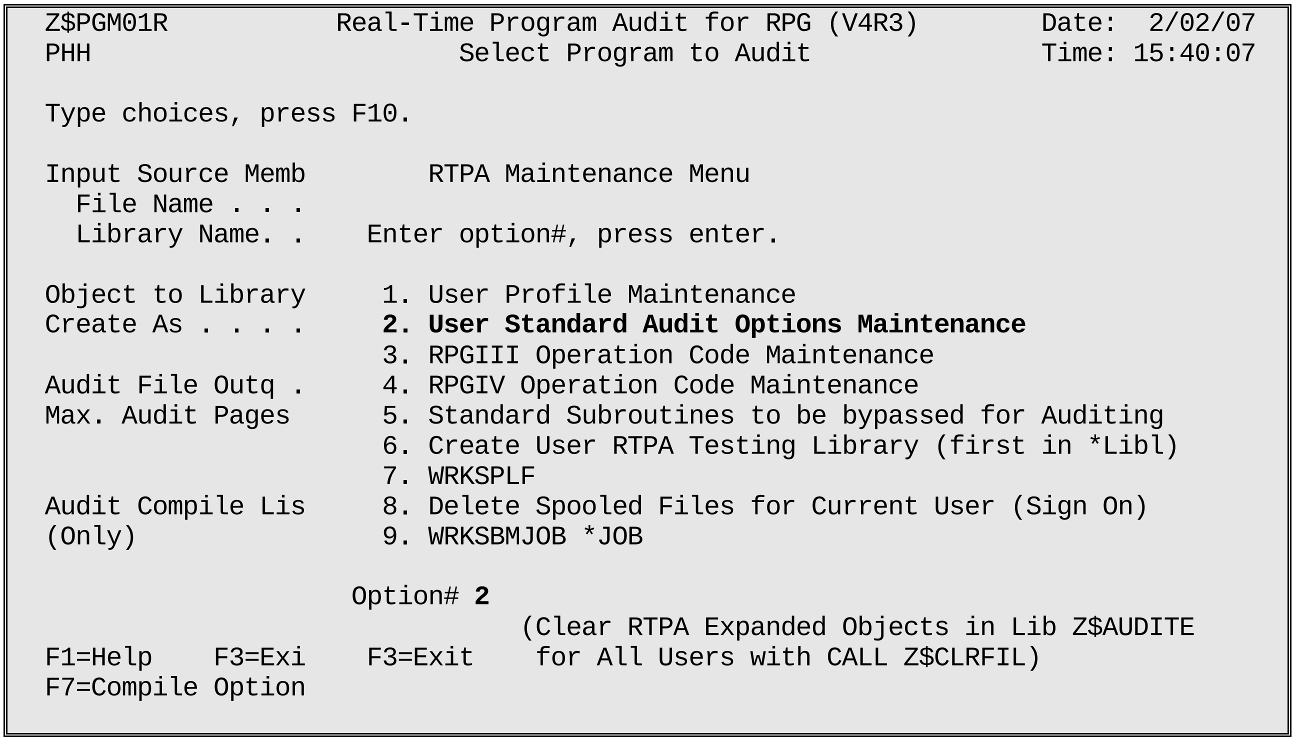
Chapter 7: **Using Auditing Options**

RTPA audit options let you control what information is captured during auditing.

Auditing Options

The User can control how RTPA expands the copy of the input source program with Z$ audit source statements by using the RTPA Audit Options.

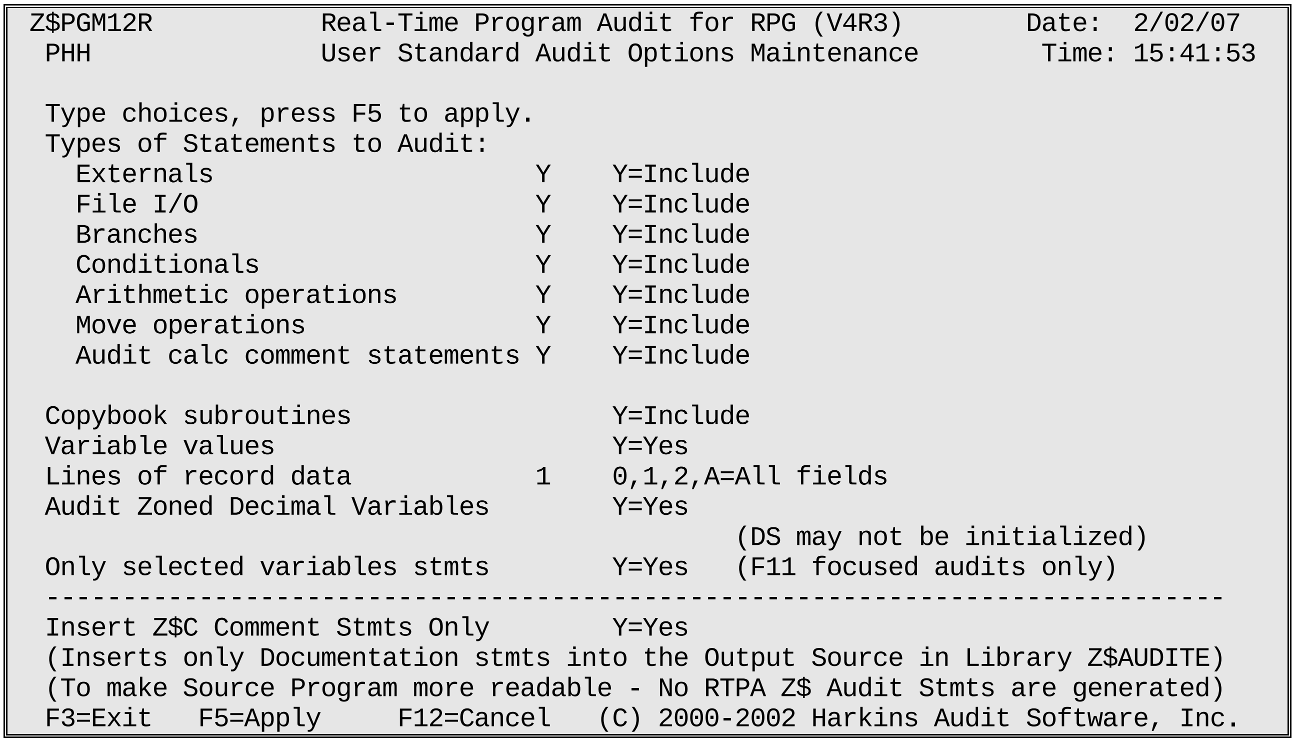
RTPA User Audit Option defaults are created dynamically when the user first signs on to RTPA and stored in RTPA file Z$FI02. These User Audit Options may be changed using option 2 of the RTPA Menu (command key 9 on the RTPA main screen).



```
Z$PGM01R          Real-Time Program Audit for RPG (V4R3)        Date:  2/02/07
PHH                       Select Program to Audit               Time: 15:40:07

Type choices, press F10.

Input Source Memb        RTPA Maintenance Menu
  File Name . . .
  Library Name. .    Enter option#, press enter.

Object to Library     1. User Profile Maintenance
Create As . . . .     2. User Standard Audit Options Maintenance
                      3. RPGIII Operation Code Maintenance
Audit File Outq .     4. RPGIV Operation Code Maintenance
Max. Audit Pages      5. Standard Subroutines to be bypassed for Auditing
                      6. Create User RTPA Testing Library (first in *Libl)
                      7. WRKSPLF
Audit Compile Lis     8. Delete Spooled Files for Current User (Sign On)
(Only)                9. WRKSBMJOB *JOB

                    Option# 2
                               (Clear RTPA Expanded Objects in Lib Z$AUDITE
F1=Help    F3=Exi    F3=Exit    for All Users with CALL Z$CLRFIL)
F7=Compile Option
```

Figure 7.1 Selection of User Audit Options maintenance to change User Audit Options

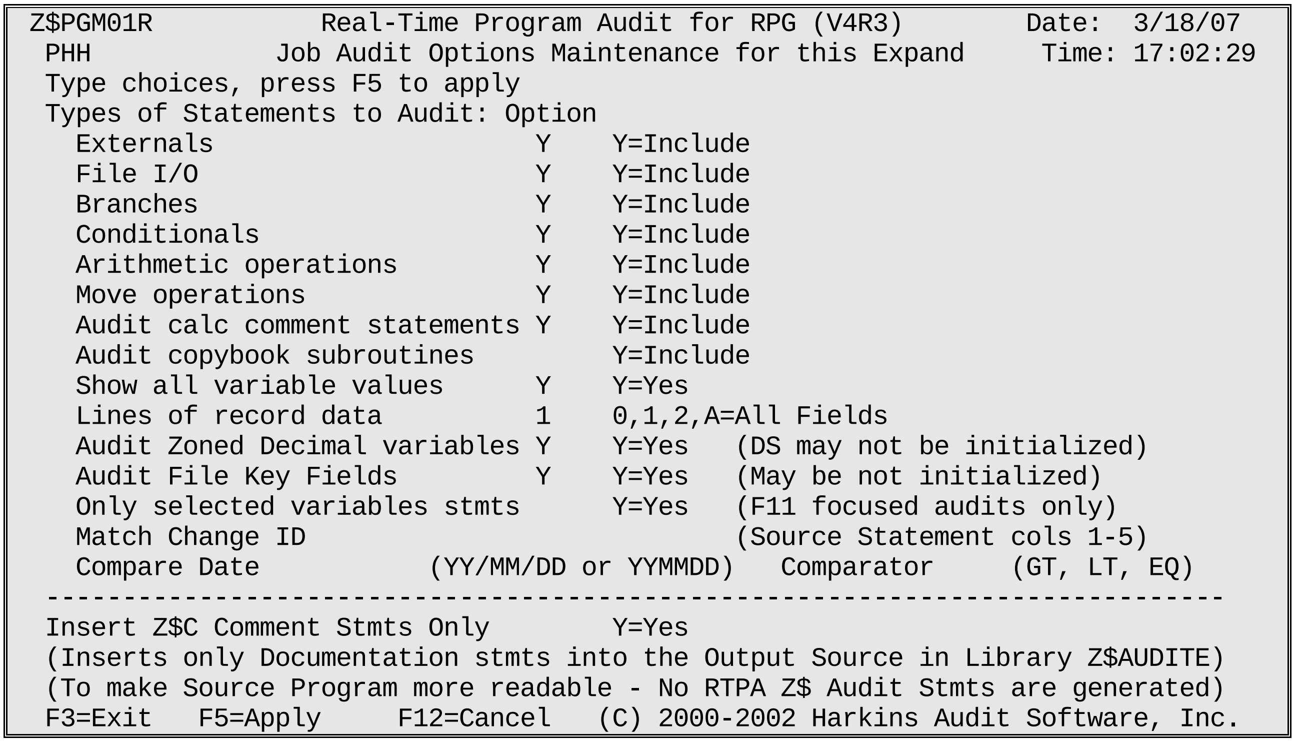
These User Audit Options are the default for all RTPA expansions for the User, unless overridden by the Command Key 6 key Job Audit Options on the RTPA main screen.



```
 Z$PGM12R          Real-Time Program Audit for RPG (V4R3)       Date:  2/02/07
  PHH              User Standard Audit Options Maintenance       Time: 15:41:53

  Type choices, press F5 to apply.
  Types of Statements to Audit:
    Externals                  Y    Y=Include
    File I/O                   Y    Y=Include
    Branches                   Y    Y=Include
    Conditionals               Y    Y=Include
    Arithmetic operations      Y    Y=Include
    Move operations            Y    Y=Include
    Audit calc comment statements Y    Y=Include

  Copybook subroutines              Y=Include
  Variable values                   Y=Yes
  Lines of record data         1    0,1,2,A=All fields
  Audit Zoned Decimal Variables     Y=Yes
                                            (DS may not be initialized)
  Only selected variables stmts     Y=Yes   (F11 focused audits only)
  -----------------------------------------------------------------------------
  Insert Z$C Comment Stmts Only     Y=Yes
  (Inserts only Documentation stmts into the Output Source in Library Z$AUDITE)
  (To make Source Program more readable - No RTPA Z$ Audit Stmts are generated)
  F3=Exit   F5=Apply    F12=Cancel   (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 7.2 RTPA User Audit Options defaults maintenance, for all expansions

4. RTPA Job Audit Options for only this RTPA expansion and are accomplished by pressing Command key 6 on the RTPA main screen.



```
Z$PGM01R          Real-Time Program Audit for RPG (V4R3)       Date:  3/18/07
 PHH            Job Audit Options Maintenance for this Expand     Time: 17:02:29
 Type choices, press F5 to apply
 Types of Statements to Audit: Option
   Externals                    Y    Y=Include
   File I/O                     Y    Y=Include
   Branches                     Y    Y=Include
   Conditionals                 Y    Y=Include
   Arithmetic operations        Y    Y=Include
   Move operations              Y    Y=Include
   Audit calc comment statements Y    Y=Include
   Audit copybook subroutines        Y=Include
   Show all variable values     Y    Y=Yes
   Lines of record data         1    0,1,2,A=All Fields
   Audit Zoned Decimal variables Y    Y=Yes   (DS may not be initialized)
   Audit File Key Fields        Y    Y=Yes   (May be not initialized)
   Only selected variables stmts      Y=Yes   (F11 focused audits only)
   Match Change ID                            (Source Statement cols 1-5)
   Compare Date           (YY/MM/DD or YYMMDD)   Comparator     (GT, LT, EQ)
 -----------------------------------------------------------------------------
 Insert Z$C Comment Stmts Only        Y=Yes
 (Inserts only Documentation stmts into the Output Source in Library Z$AUDITE)
 (To make Source Program more readable - No RTPA Z$ Audit Stmts are generated)
 F3=Exit   F5=Apply     F12=Cancel   (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 7.3 RTPA Job Audit Options overrides for this expansion only

- **Press F5** to apply the Job audit options overrides, then F10 or F11 to submit the RTPA expansion

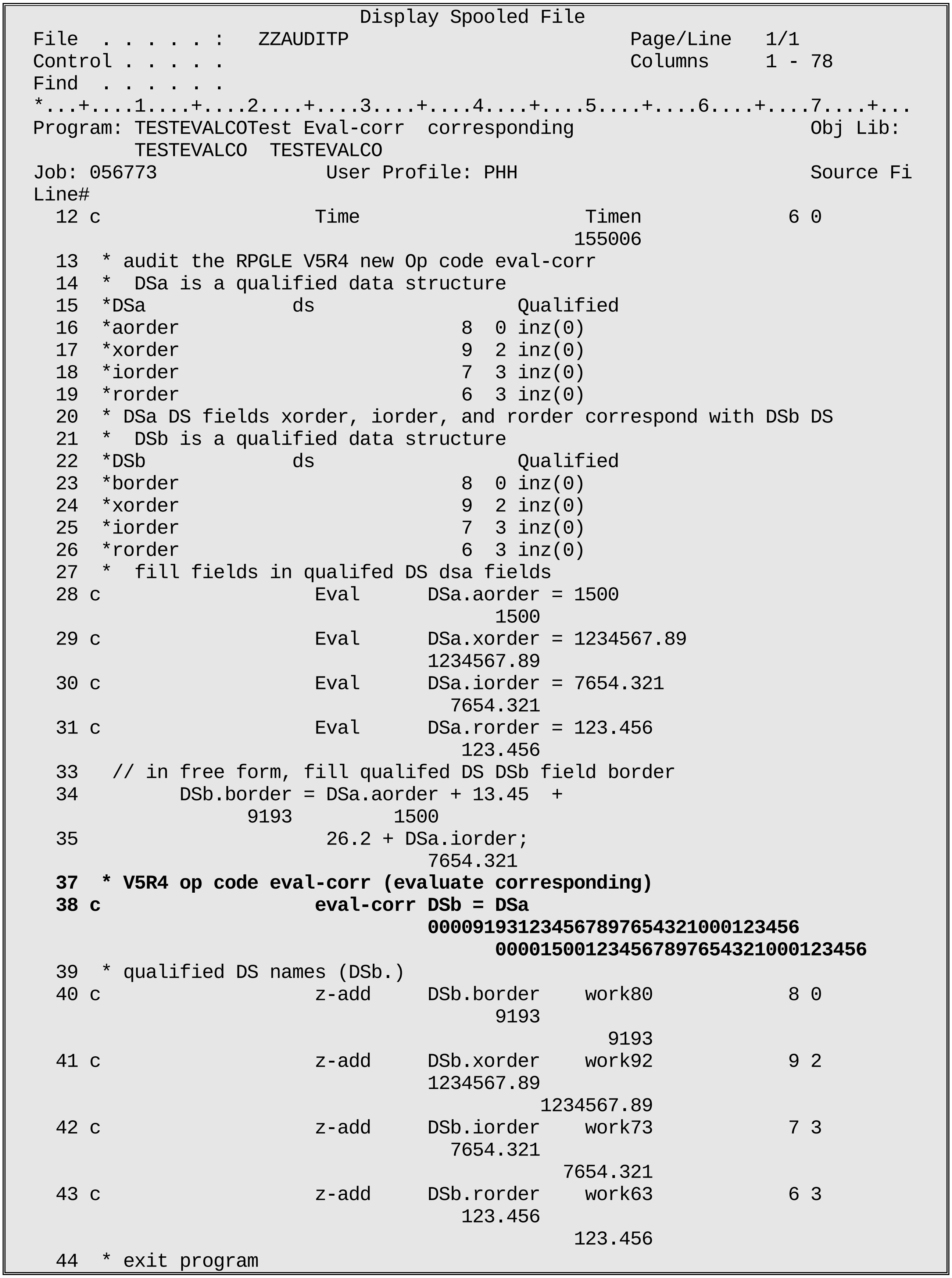
Note: Programmer default Audit Options and overrides (for all RTPA expands) are maintained using the Option 2 of the RTPA Maintenance Menu.

The first set of options allow you to include or exclude certain operations. For your convenience, the operations have been grouped as follows:

Arithmetic	Branch	Conditional	External	File I/O	Move
ADD	BEGSR	CHECK	CALL	CHAIN	CAT
ADDDUR	CAB	CHECKR	CALLB	CLOSE	CLEAR
DIV	CAS	COMP	CALLP	DELETE	MOVE
EVAL	ENDSR	DO	IN	EXCEPT	MOVEAX
EXTRCT	EXSR	DOU	OUT	EXFMT	RESET
MULT	GOTO	DOW	PARM	OPEN	SETOFF
MVR	TAG	ELSE	RETURN	READ	SETON
SUB		END		READPE	SUBST
SUBDUR		ENDFOR		SET	XLATE
TIME		ENDSL		UNLOCK	
XFOOT		FOR		UPDATE	
Z-ADD		IF		WRITE	
Z-SUB		ITER			
EVALR		LEAVE			
EVAL-CORR		LEAVESR			
		LOOKUP			
		MONITOR			
		OCCUR			
		ON-ERROR			
		OR			
		OTHER			
		SCAN			
		SELECT			
		TEST			
		TESTN			
		WH			
		WHEN			

In addition, virtually all IBM RPG V5R3 and V5R4 executable operation codes and BIFs are audited by RPTA for RPG.

For example, the V5R4 operation code eval-corr



```
                             Display Spooled File
File  . . . . . :   ZZAUDITP                        Page/Line   1/1
Control . . . . .                                   Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
Program: TESTEVALCOTest Eval-corr  corresponding                    Obj Lib:
         TESTEVALCO  TESTEVALCO
Job: 056773              User Profile: PHH                          Source Fi
Line#
  12 c                   Time                    Timen             6 0
                                                155006
  13  * audit the RPGLE V5R4 new Op code eval-corr
  14  *  DSa is a qualified data structure
  15  *DSa             ds                  Qualified
  16  *aorder                        8  0 inz(0)
  17  *xorder                        9  2 inz(0)
  18  *iorder                        7  3 inz(0)
  19  *rorder                        6  3 inz(0)
  20  * DSa DS fields xorder, iorder, and rorder correspond with DSb DS
  21  *  DSb is a qualified data structure
  22  *DSb             ds                  Qualified
  23  *border                        8  0 inz(0)
  24  *xorder                        9  2 inz(0)
  25  *iorder                        7  3 inz(0)
  26  *rorder                        6  3 inz(0)
  27  *  fill fields in qualifed DS dsa fields
  28 c                   Eval      DSa.aorder = 1500
                                         1500
  29 c                   Eval      DSa.xorder = 1234567.89
                                   1234567.89
  30 c                   Eval      DSa.iorder = 7654.321
                                     7654.321
  31 c                   Eval      DSa.rorder = 123.456
                                      123.456
  33   // in free form, fill qualifed DS DSb field border
  34        DSb.border = DSa.aorder + 13.45  +
                  9193         1500
  35                      26.2 + DSa.iorder;
                                   7654.321
  37  * V5R4 op code eval-corr (evaluate corresponding)
  38 c                   eval-corr DSb = DSa
                                   000091931234567897654321000123456
                                         000015001234567897654321000123456
  39  * qualified DS names (DSb.)
  40 c                   z-add     DSb.border    work80            8 0
                                         9193
                                                   9193
  41 c                   z-add     DSb.xorder    work92            9 2
                                   1234567.89
                                             1234567.89
  42 c                   z-add     DSb.iorder    work73            7 3
                                     7654.321
                                               7654.321
  43 c                   z-add     DSb.rorder    work63            6 3
                                      123.456
                                                123.456
  44  * exit program
```

```
    45 c                   Eval      *inlr = *on
                                     1
    46 c                   Return
```

Figure 7.4 RTPA audit of R5R4 Operation code eval-corr (evaluate corresponding)
(Only the leftmost 80 characters of the ZZAUDITP audit output are shown)

Pre-Audit Conditionals

Conditionals are audited after they are executed. As a result, conditionals that evaluate as "not true" are not included in the audit file. For that reason, selecting Y for Pre-Audit Conditionals causes RTPA to also audit conditionals prior to evaluation. The result is that false conditionals are displayed.

Arithmetic Operations

Arithmetic operations are audited after the entire expression is executed. If a variable value is changed during the expression, the changed value is shown. For example:

```
0101 C     EVAL Result = 0
0102 C     EVAL Result = Result + 10
```

Produces the audit fragment:

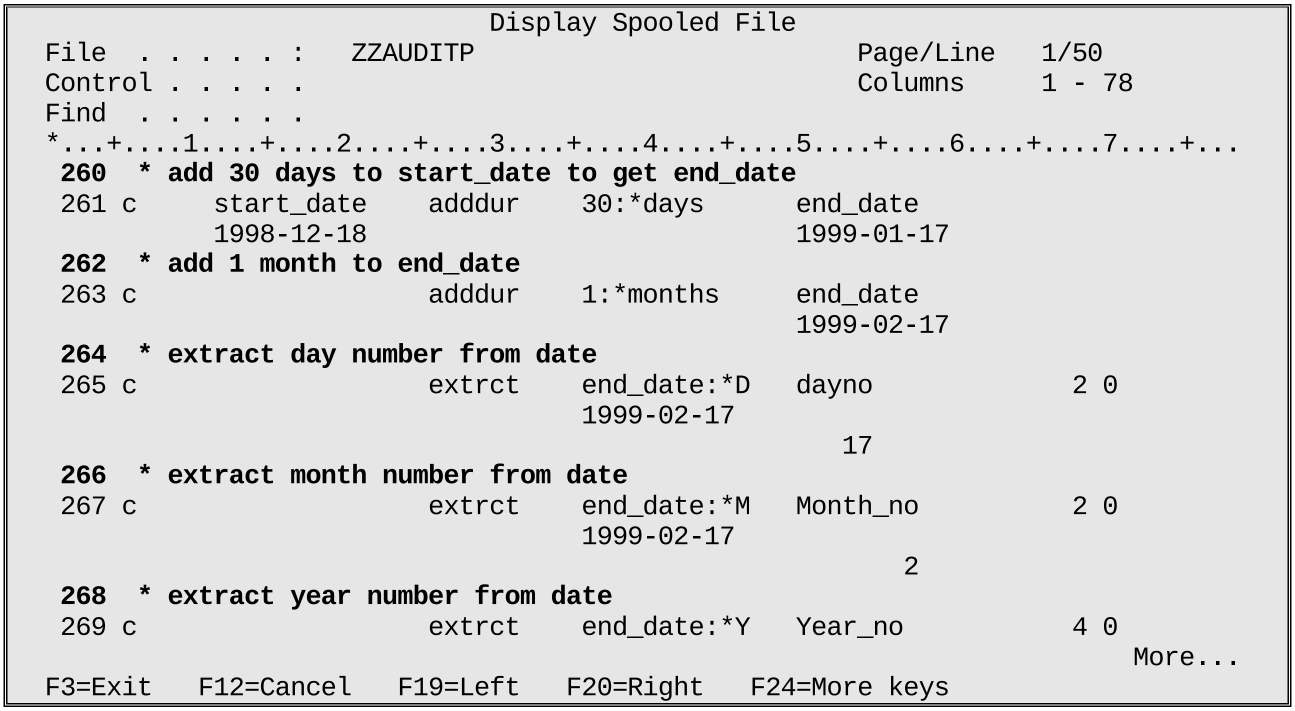
```
   200C     EVAL Result = 0
                     0
   201C     EVAL Result = Result + 10
                    10       10
```

Auditing Calculation Comment statements

RTPA always shows comments that are on audited source lines (as part of the audited source statement).. Optionally, comment statements on separate calculation source statements may be included in the RTPA audit output. The RTPA default is to audit calculations comment statements .Including comments in the audit output may help in understanding executing code, particularly if the RPG code is generated from a code generator. Each audited calculation comment statement required four Z$ audit source statements in the expanded source program in library Z$AUDITE.

The default option is to audit separate calculation comments.

```
Audit calc comment statements Y    Y=Include
```



```
                             Display Spooled File
File  . . . . . :   ZZAUDITP                         Page/Line   1/50
Control . . . . .                                    Columns     1 - 78
Find  . . . . . .
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 260  * add 30 days to start_date to get end_date
 261 c     start_date    adddur    30:*days      end_date
           1998-12-18                            1999-01-17
 262  * add 1 month to end_date
 263 c                   adddur    1:*months     end_date
                                                 1999-02-17
 264  * extract day number from date
 265 c                   extrct    end_date:*D   dayno            2 0
                                   1999-02-17
                                                    17
 266  * extract month number from date
 267 c                   extrct    end_date:*M   Month_no         2 0
                                   1999-02-17
                                                        2
 268  * extract year number from date
 269 c                   extrct    end_date:*Y   Year_no          4 0
                                                                    More...
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 7.5 RTPA auditing of calculations comment statements

To be included in the audit, the comments must occur in the source inside the execution flow (i.e., between a beginning operation such as TAG and an end operation such as ENDSR).

Show All Variable Values

This causes the audit file to include variable contents for all variables in executed statements.

Audit lines of data record

This tells RTPA how many 198 character lines of a data record should be included in an audit file. The name of each data field used in the record is output, followed by a dash, followed by the contents of the field.

Only Selected Variable Statements

This is used only for Advanced Auditing (F11) functionality, and only displays the data contents for variables selected with a Y (F16 to display the variables or fields used in the program).

Audit Zoned Decimal Variables

This is used to audit the contents of variables defined as a DS (Defined storage) in the D specifications. Y is the default option. Numeric DS fields may contain non-numeric data if not properly initialized (INT), or filled with proper numeric data before being used in the program.

Audit File Key Fields

This is used to audit the contents of the File Key fields for all File I/O operations. The default is Y.

QSYSPRT Compile Printer File

THE RPG compile output for the input RPG or RPGLE source program and the expanded compile with RTPA Z$ statements is on printer file QSYSPRT. The QSYSPRT printer file should have a maximum records of at least 600,000 records to allow for very large compile listing output of expanded programs.

Auditing by Change ID

You can choose to audit only those source lines which have a certain Change ID.

- On the Job audit options screen (F6), tab to `Match Change ID.` Enter the Change ID that you want to audit.

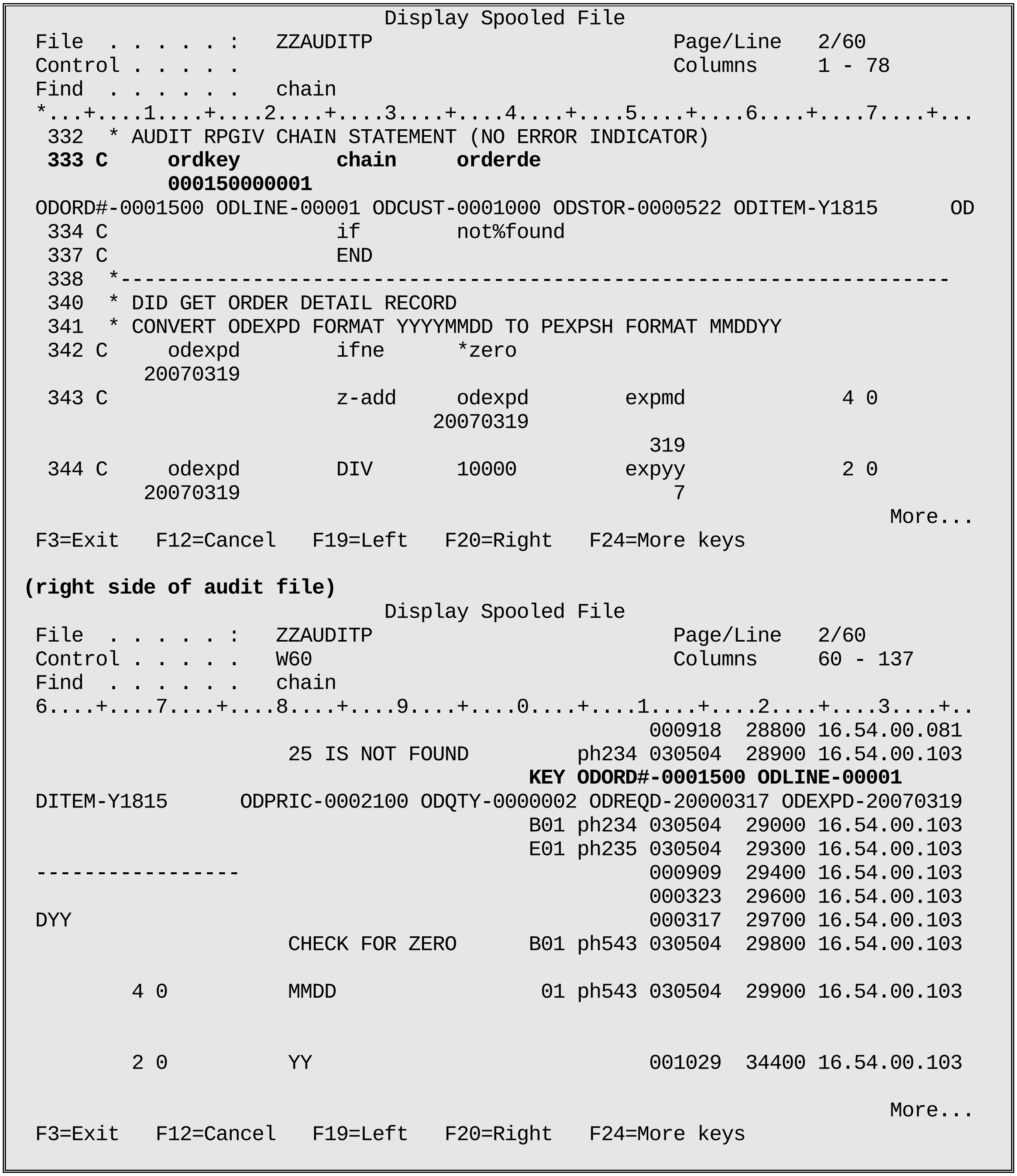
Auditing by Change Date

You can choose to limit your audit to source lines based on their change date.

- On the main screen, tab to `Compare Date.` Enter the date.

- Tab to `Comparator` and enter GT (Greater Than), LT (Less Than) or EQ (Equal to).

The audit will only include source lines whose Change Date compares correctly to the Comparator

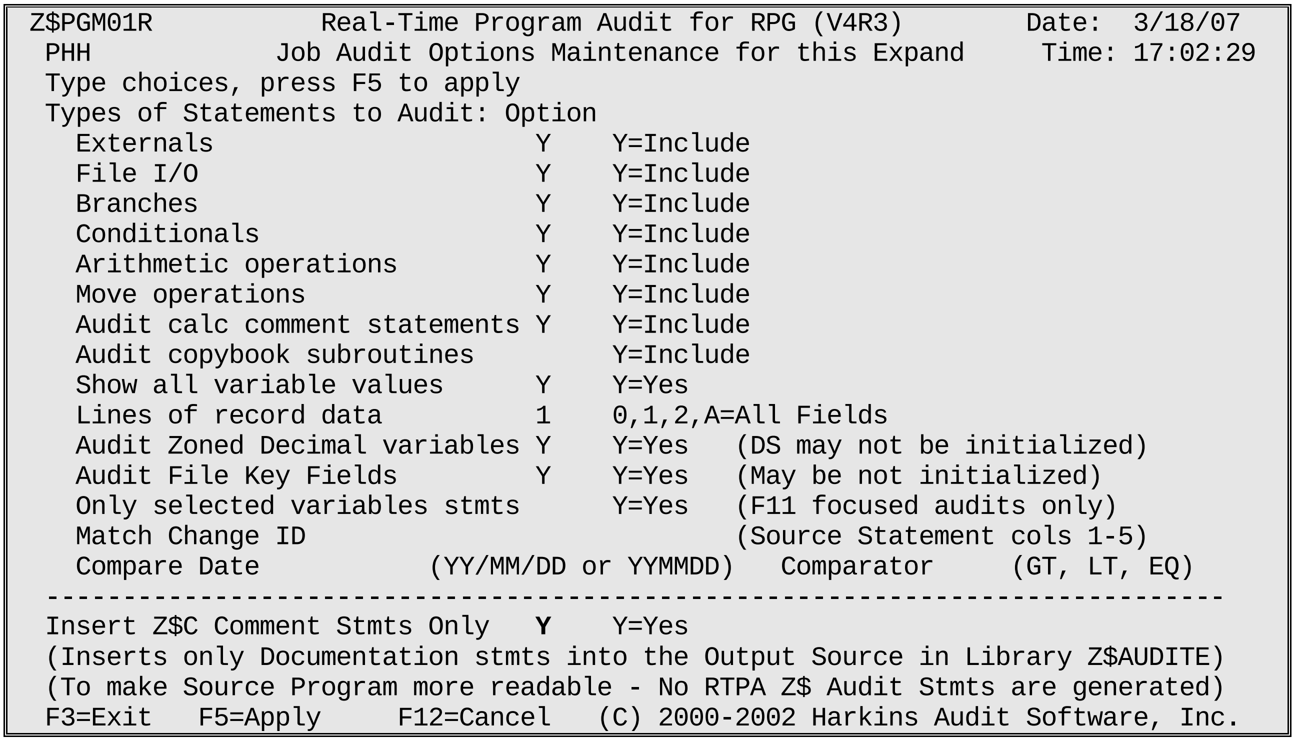


```
                              Display Spooled File
 File  . . . . . :   ZZAUDITP                         Page/Line   2/60
 Control . . . . .                                    Columns     1 - 78
 Find  . . . . . .   chain
 *...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
  332  * AUDIT RPGIV CHAIN STATEMENT (NO ERROR INDICATOR)
  333 C     ordkey        chain     orderde
            000150000001
 ODORD#-0001500 ODLINE-00001 ODCUST-0001000 ODSTOR-0000522 ODITEM-Y1815      OD
  334 C                   if        not%found
  337 C                   END
  338  *---------------------------------------------------------------------
  340  * DID GET ORDER DETAIL RECORD
  341  * CONVERT ODEXPD FORMAT YYYYMMDD TO PEXPSH FORMAT MMDDYY
  342 C     odexpd        ifne      *zero
          20070319
  343 C                   z-add     odexpd        expmd             4 0
                                  20070319
                                                    319
  344 C     odexpd        DIV       10000         expyy             2 0
          20070319                                    7
                                                                        More...
 F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys

(right side of audit file)
                              Display Spooled File
 File  . . . . . :   ZZAUDITP                         Page/Line   2/60
 Control . . . . .   W60                              Columns     60 - 137
 Find  . . . . . .   chain
 6....+....7....+....8....+....9....+....0....+....1....+....2....+....3....+..
                                                    000918  28800 16.54.00.081
                      25 IS NOT FOUND         ph234 030504  28900 16.54.00.103
                                          KEY ODORD#-0001500 ODLINE-00001
 DITEM-Y1815      ODPRIC-0002100 ODQTY-0000002 ODREQD-20000317 ODEXPD-20070319
                                          B01 ph234 030504  29000 16.54.00.103
                                          E01 ph235 030504  29300 16.54.00.103
 -----------------                                  000909  29400 16.54.00.103
                                                    000323  29600 16.54.00.103
 DYY                                                000317  29700 16.54.00.103
                      CHECK FOR ZERO      B01 ph543 030504  29800 16.54.00.103

         4 0          MMDD                 01 ph543 030504  29900 16.54.00.103


         2 0          YY                            001029  34400 16.54.00.103

                                                                        More...
 F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure 7.6 Example of auditing File Key fields for a CHAIN operation

Documentation Only with Z$C Comment Auditing

The User can insert RTPA documentation comment statements to make the copied input source statement more readable. The RTPA Documentation Only option only inserts documentation Z$ comment statements, and no executable Z$ audit statements are inserted with this option.

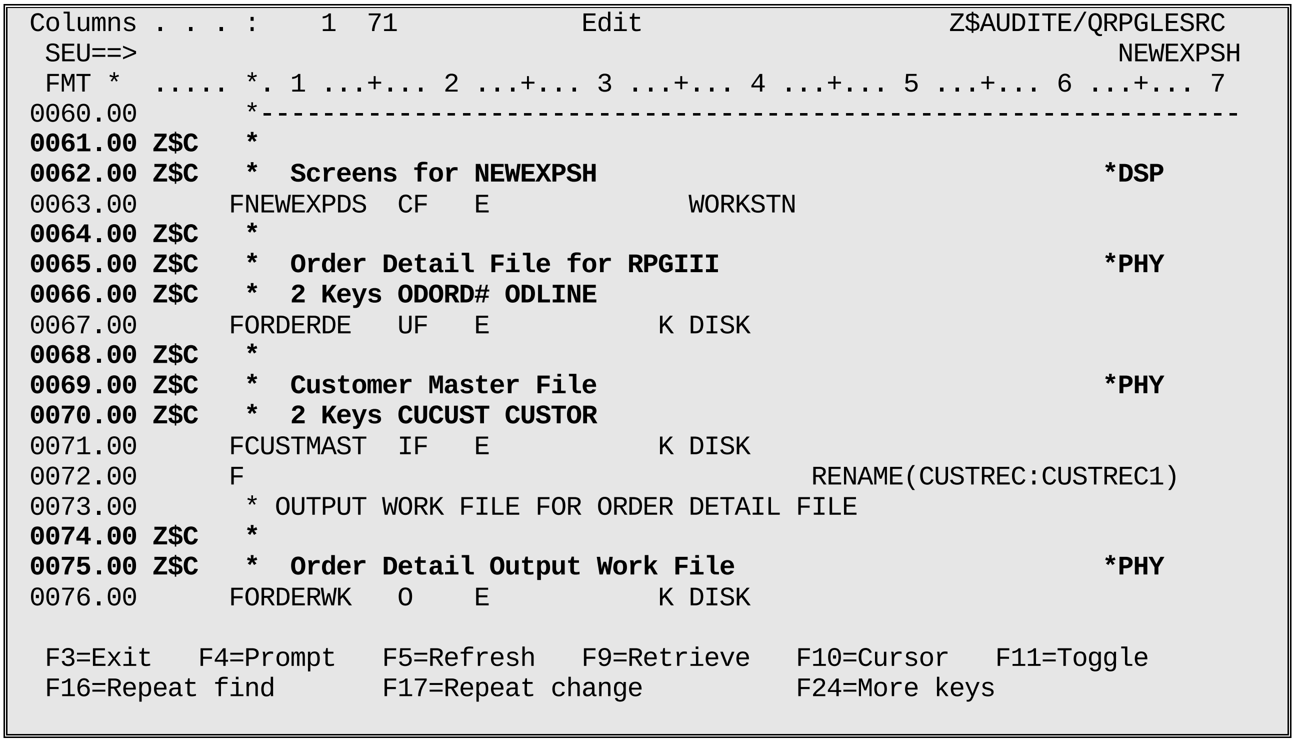


```
Z$PGM01R           Real-Time Program Audit for RPG (V4R3)        Date:  3/18/07
 PHH            Job Audit Options Maintenance for this Expand     Time: 17:02:29
 Type choices, press F5 to apply
 Types of Statements to Audit: Option
   Externals                     Y    Y=Include
   File I/O                      Y    Y=Include
   Branches                      Y    Y=Include
   Conditionals                  Y    Y=Include
   Arithmetic operations         Y    Y=Include
   Move operations               Y    Y=Include
   Audit calc comment statements Y    Y=Include
   Audit copybook subroutines         Y=Include
   Show all variable values      Y    Y=Yes
   Lines of record data          1    0,1,2,A=All Fields
   Audit Zoned Decimal variables Y    Y=Yes   (DS may not be initialized)
   Audit File Key Fields         Y    Y=Yes   (May be not initialized)
   Only selected variables stmts      Y=Yes   (F11 focused audits only)
   Match Change ID                            (Source Statement cols 1-5)
   Compare Date           (YY/MM/DD or YYMMDD)   Comparator     (GT, LT, EQ)
 -----------------------------------------------------------------------------
 Insert Z$C Comment Stmts Only   Y    Y=Yes
 (Inserts only Documentation stmts into the Output Source in Library Z$AUDITE)
 (To make Source Program more readable - No RTPA Z$ Audit Stmts are generated)
 F3=Exit   F5=Apply     F12=Cancel   (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 7.7 Selection of RTPA documentation only Z$C statements into the copied source

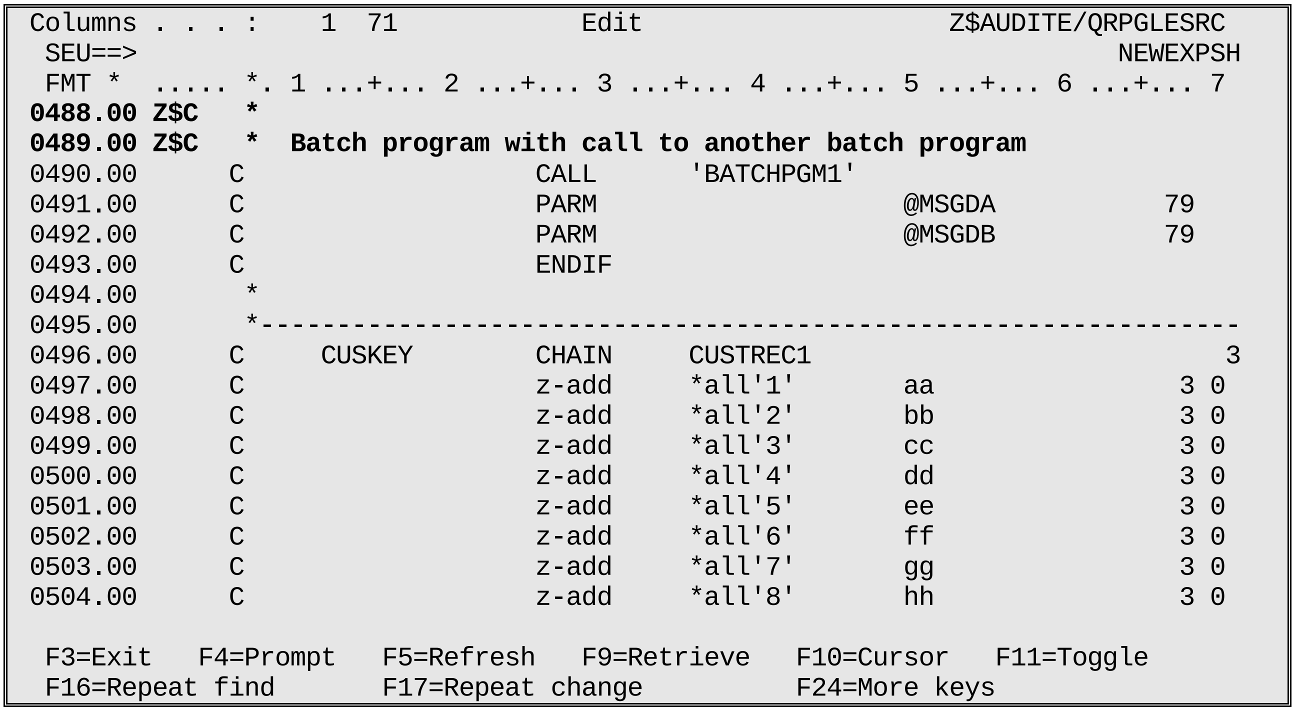
The copied source program in library Z$AUDITE contains the input source program statements, and Z$C comment statements which document the source program, including:

- File descriptions and the file keys
- Called program descriptions



```
Columns . . . :    1  71            Edit                    Z$AUDITE/QRPGLESRC
 SEU==>                                                                NEWEXPSH
 FMT *  ..... *. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0060.00       *-----------------------------------------------------------------
0061.00 Z$C   *
0062.00 Z$C   *  Screens for NEWEXPSH                                 *DSP
0063.00      FNEWEXPDS  CF   E             WORKSTN
0064.00 Z$C   *
0065.00 Z$C   *  Order Detail File for RPGIII                         *PHY
0066.00 Z$C   *  2 Keys ODORD# ODLINE
0067.00      FORDERDE   UF   E           K DISK
0068.00 Z$C   *
0069.00 Z$C   *  Customer Master File                                 *PHY
0070.00 Z$C   *  2 Keys CUCUST CUSTOR
0071.00      FCUSTMAST  IF   E           K DISK
0072.00      F                                     RENAME(CUSTREC:CUSTREC1)
0073.00       * OUTPUT WORK FILE FOR ORDER DETAIL FILE
0074.00 Z$C   *
0075.00 Z$C   *  Order Detail Output Work File                        *PHY
0076.00      FORDERWK   O    E           K DISK

 F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
 F16=Repeat find       F17=Repeat change          F24=More keys
```

Figure 7.8 RTPA documentation only Z$C File description comments



```
Columns . . . :    1  71            Edit                    Z$AUDITE/QRPGLESRC
 SEU==>                                                                NEWEXPSH
 FMT *  ..... *. 1 ...+... 2 ...+... 3 ...+... 4 ...+... 5 ...+... 6 ...+... 7
0488.00 Z$C   *
0489.00 Z$C   *  Batch program with call to another batch program
0490.00      C                   CALL      'BATCHPGM1'
0491.00      C                   PARM                    @MSGDA           79
0492.00      C                   PARM                    @MSGDB           79
0493.00      C                   ENDIF
0494.00       *
0495.00       *-----------------------------------------------------------------
0496.00      C     CUSKEY        CHAIN     CUSTREC1                           3
0497.00      C                   z-add     *all'1'       aa                3 0
0498.00      C                   z-add     *all'2'       bb                3 0
0499.00      C                   z-add     *all'3'       cc                3 0
0500.00      C                   z-add     *all'4'       dd                3 0
0501.00      C                   z-add     *all'5'       ee                3 0
0502.00      C                   z-add     *all'6'       ff                3 0
0503.00      C                   z-add     *all'7'       gg                3 0
0504.00      C                   z-add     *all'8'       hh                3 0

 F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
 F16=Repeat find       F17=Repeat change          F24=More keys
```

Figure 7.9 RTPA documentation only Z$C Called program description comments

The RTPA inserted Z$C comment statements can be very helpful to a programmer when reviewing an unfamiliar large program. No RTPA Z$ audit statements are inserted in this document only RTPA expansion.

Chapter

8

Chapter 8: **Auditing Very Large RPG and COBOL Programs**

This chapter provides guidance on how to get around the size limitations of the RPGIII compiler and the COBOL/400 compiler when auditing very large programs.

RTPA audit-enables source files by creating new source files with additional audit statements. These new source files can be 3-5 times larger than the original source files. Thus, a source program of 5,000 source statements (including copy books and external file definitions) may produce an expanded source file with more than 25,000 lines of source code. (It is worth noting that these additional instructions generally have a smaller, non-proportional impact on execution time.)

RPG Compiler Limits

The RPGIII compiler has design constraints that limit a typical RPGIII source program to approximately 20,000 to 25,000 source statements. Larger RPG programs will cause a compiling error.

The RPGIV (ILE) compiler has design constraints that limit a typical RPG ILE source program to approximately 60,000 to 65,000 source statements. Larger RPG programs may cause a compiling error.

The actual number of RPG source statements allowed for a successful compile of an RPGIII or RPGIV source program depends on the design constraints of each phase of the RPG compiler, and the source statements used in the source program.

RPGIII source programs and RPGIII copy books may be converted to RPGIV to use the RPG ILE Compiler with the IBM Command CVTRPGSRC.

COBOL Compiler Limits

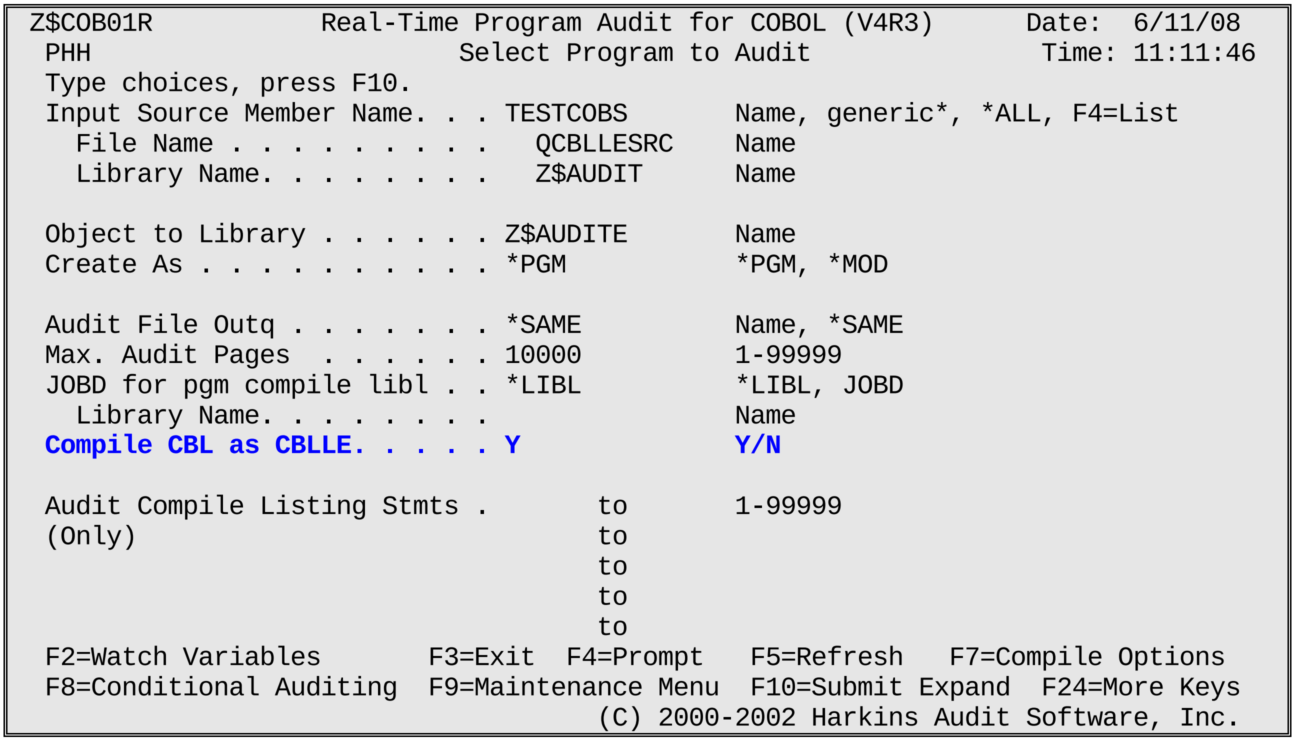
The COBOL/400 compiler has design constraints that limit a typical COBOL/400 source program to approximately 20,000 to 25,000 source statements. Larger COBOL programs will cause a compiling error.

The COBOL ILE compiler has design constraints that limit a typical COBOL ILE source program to approximately 60,000 to 65,000 source statements. Larger COBOL ILE RPG programs may cause a compiling error.

The actual number of COBOL source statements allowed for a successful compile of an COBOL/400 or COBOL ILE source program depends on the design constraints of each phase of the COBOL compiler, and the source statements used in the source program.

RTPA for COBOL provides the capability to expand COBOL/400 source programs (Source type CBL and SQLCBL) as COBOL ILE programs (source type CBLLE and SQLCBLLE) automatically by entering a Y in the Compile CBL as CBLLE option (without changing the input program source type).

```
Compile CBL as CBLLE. . . . . Y      Y/N
```



```
Z$COB01R          Real-Time Program Audit for COBOL (V4R3)      Date:  6/11/08
 PHH                        Select Program to Audit               Time: 11:11:46
 Type choices, press F10.
 Input Source Member Name. . . TESTCOBS       Name, generic*, *ALL, F4=List
   File Name . . . . . . . . .   QCBLLESRC    Name
   Library Name. . . . . . . .   Z$AUDIT      Name

 Object to Library . . . . . . Z$AUDITE       Name
 Create As . . . . . . . . . . *PGM           *PGM, *MOD

 Audit File Outq . . . . . . . *SAME          Name, *SAME
 Max. Audit Pages  . . . . . . 10000          1-99999
 JOBD for pgm compile libl . . *LIBL          *LIBL, JOBD
   Library Name. . . . . . . .                Name
 Compile CBL as CBLLE. . . . . Y              Y/N

 Audit Compile Listing Stmts .       to       1-99999
 (Only)                              to
                                     to
                                     to
                                     to
 F2=Watch Variables       F3=Exit  F4=Prompt   F5=Refresh   F7=Compile Options
 F8=Conditional Auditing  F9=Maintenance Menu  F10=Submit Expand  F24=More Keys
                                     (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure 8.1 Compiling a COBOL/400 source program as a COBOL ILE Source program

SEU Limits

The Source Entry Utility (SEU) editor currently has a design limit of approximately 32,764 source statements for an RPG source program. This means that RTPA expanded source programs may not be edited with SEU if the expanded source program exceeds 32,764 source statements. Other editors or disk file utility programs may be used to edit source members which are larger than the 32,764 statements limit.

How RTPA Inserts Audit Statements

RTPA examines every executable calculation statement in the input source program to determine if the statement and its variables should be audited (based on the auditing options that the programmer selected).

RTPA defaults to insert audit statements in the expanded source program to provide audit output of both the RPG calculation statements, and the data contents of all variables used in the calculation statements. In addition, RTPA defaults to display one line of 132 characters of the fields in records processed, and the contents of parameters and data areas used during execution of the expanded object program (which is created from the expanded source program).

Very large source programs that cannot be successfully completely expanded with RTPA audit statements of all executing source statements may be selectively audited by specifying ranges of from and to compile listing statements to be audited.

Changing Audit Options to Reduce the Source Size

You may estimate that about eight to nine Z$ audit statements are required for each line of calculation statements to be completely audited, including the complete source statement and the data contents of all variables in the statement.

Simply bypassing the auditing of copybooks and subroutines designated as utility subroutines dramatically reduces the number of audit statements inserted and produces more focused audit output.

Audit Copybook Subroutines
Blank this option to not audit copybook subroutines. Many programs use copybooks to copy standard subroutines into the program. These copied subroutines can most often be bypassed for auditing, resulting in more focused auditing of the code of interest to the programmer.

Audit lines of Record Data
Enter 0 (zero) in this option to bypass the auditing of the data contents of records as they are processed. The contents of the variables (fields) in the records will still be audited as the calculations statements using those variables are executed.

Reporting RTPA Status 9 Compile Error

If the RTPA for RPG expanded source compile results in a status 9 compile error (rather than the status 8 OK), then RTPA for RPG may have incorrectly inserted the Z$ audit statements into the copy of your input source statement program. In that case, please copy the expanded source compile listing from your spool file into a Microsoft Notepad (Microsoft, Accessories, Notepad) and email the saved Notepad document to Harkins Audit Software, Inc. as an email attachment, with a brief description of the problem.

Note - D Specification copy books must use D/COPY (not F/COPY or I/COPY)

Summary

Because of current compiler design constraints, auditing very large RPG source programs may require knowledge of how RTPA audits are accomplished, and thoughtful selection of the appropriate RTPA audit options to be used to expand the input source program for RTPA testing.

RTPA provides the capability of many auditing options to allow the programmer to set appropriate default options for virtually any programming need. Auditing options in the may be changed to accomplish auditing of very large programs.

Conversion of very large RPGIII source programs to RPGIV source programs with the CVTRPGSRC command allows much more extensive RTPA auditing, particularly if no /COPY copybooks are used in the source programs. (The RPGIII /COPY copybooks would also have to be converted to RPGIV.)

Appendix

A

Appendix A: **Frequently Asked Questions**

Does RTPA for RPG audit all RPG programs?

RTPA for RPG is designed to audit all types RPG source programs including: RPG, RPT, SQLRPG, RPGMOD, RPGLE, SQLRPGLE and SQLRPGMOD.

RTPA for RPG will not work on extremely large programs that are very close to the maximum number of lines that the IBM RPG compiler will accept. See **Chapter 8: Auditing Very Large RPG Programs** for more information. The RPG ILE compiler does compile large programs, but still has some compiler limits primarily due to the half-word (32K) limits on strings of elements in the compiler. The same is true with the limit of 32K statements for the source program member and some editors (such as SEU), which could otherwise support 99,999 source statements, without these design limitations. 4GL languages and code generators such as RTPA for RPG typically generate very large source programs.

Does RTPA audit freeform RPG?

Yes, RTPA audits freeform (free-format) RPG as in the following example:

```
...Source Code Fragment...
0100.00 /free
0101.00   torder = 1500;
0102.00   iorder = 78.543;
0103.00   xorder = torder + 13.45  +
0104.00            26.2 + iorder;
0105.00 /end-free
...
```

```
...Audit File Fragment...
0201   torder = 1500;
         1500
0202   iorder = 78.543;
       78.543
0203   xorder = torder + 13.45  +
      1618.19     1500
0204            26.2 + iorder;
                       78.543
...
```

Why did RTPA fail to expand the program correctly?

Please make sure that the program compiles properly prior to submitting the program to RTPA. RTPA uses the AS/400's native compiler – if the program won't compile without RTPA, it won't compile under RTPA.

If your program compiles normally but does not work properly with RTPA, please contact technical support.

Why can't I compile my large RPGIII program?

The RPGIII compiler is limited to about 20,000 source statements. Because RTPA expands the source code, the resulting source code may exceed the limits of the RPG compiler. In that case, reduce the amount of information selected for audit in the RTPA Expansion Utility.

Can I ship expanded object programs to other computers?

RTPA for RPG expanded (audit enabled) object programs may execute on another System I computer which does not have the RTPA software installed on it. However, the RTPA audit output Printer File ZZAUDITP must be created on the computer which executes the RTPA expanded object program and produces the RTPA audit output..

RTPA for RPG is intended for use in program development, unit testing, system testing, pilot production, and in production problem analysis and correction. RTPA is not intended for use in normal production processing, except as needed. You may not leave RTPA code in any software that you sell or license to any third party.

Does RTPA for RPG audit copybook statements?

RTPA audits copybook statements by default. If you do not want to audit copybook statements, you can disable auditing of copybook statements in the auditing options.

RTPA does not expand I/COPY copybook statements. For copybooks with D specs, such as SDS definitions, use the format D/COPY instead of I/COPY (as in RPGIII).

Will RTPA exceed maximum file limits in RPGIII?

RTPA increases the number of files used in the RPG source program by one. (This is the printer file on which the audit file is created.) If the program has already opened the maximum number of files allowed in RPGIII, this will cause the maximum number of files allowed in the program to be exceeded.

Does RTPA for RPG use any indicators?

RTPA for RPG uses the first unused indicator in the source program to turn auditing on or off. If there are no unused indicators in the source program, RTPA auditing is always on.

Does RTPA change the original source or object program?

No. RTPA creates the expanded source program in a temporary directory and puts audit-enabled object program in whatever library you specify. The RTPA User should not replace a production library object program during th RTPA expand process.

How do I expand and create Module objects?

RTPA will create Module objects rather than program objects if the Source type is RPGMOD or SQLRPGMOD. Also RTPA will create Module objects rather than Program objects if *MOD is specified on the RTPA main selection screen.
Create As . . . . ***MOD** *PGM, *MOD

Where is the audit output sent during program execution?

By default, RTPA sends its audit output to the printer file ZZAUDITP, which is placed in QGPL when RTPA is installed. This printer file must be in the library list when executing an audit-enabled object. The ZZAUDITP spool file for each audited job is placed in the spool file of the user, and may be displayed with the WRKSPLF command.
RTPA audit output may be placed in a desired output queue as the program executes by entering the output queue on the main RTPA selection screen
Audit File Outq . . . . *SAME Name, *SAME

How can I direct RTPA audit output to a specific Outq?

Enter the output queue name on the main RTPA selection screen as below:
Audit File Outq . . . . *SAME Name, *SAME

This is particularly helpful when the programmer wants to send the ZZAUDITP audit output to his or her programmer outq when the application program spooled output goes to another outq.

How can I expand all the members in a source file?

Use the mass RTPA expansion function on the main RTPA selection screen as below:
Member Name . . . . *ALL Name, generic*, *ALL, F4 List
*ALL in the program name and pressing Command Key 10 expands all source programs in the source file and library.

Member Name . . . . TEST* Name, generic*, *ALL, F4 List
Generic (Partial program name followed by *) expands all source programs with that name
TEST* in the program name and pressing Command Key 10 expands all source programs starting with TEST.

Do I need to expand all my source programs?

Each RPG source program that you want to audit must be expanded with the RTPA. Only expanded programs will produce audit files.

Use the RTPA generic program name (partial name suffixed with an *) to expand all RPG programs with that partial name (for example TEST* expands all programs starting with TEST).

Use the RTPA generic program name *ALL to expand **ALL** RPG programs in the source member and source file.

How can I selectively audit ranges of source statements?

Use the input compile listing compile statement numbers from the RTPA compile of the input source program to enter from statement and to statement ranges to be audited on the RTPA main screen. This is available by pressing Command Key 18 to review RTPA compile listings of both the input source program and the RTPA enabled expanded source program.

```
Audit Compile Listing Statements  2500 to  3600   1-99999
                                 4000 to  4500
                                 6001 to  6600
                                         to
                                         to
```

What is the proper format of a Compile time Array header?

**CTDATA and the name of the Compile Time Array or Table starting in position 10. No other information is valid, including comments.
**CTDATA TAXTAB

Appendix B

The RTPA Expansion Utility uses the following command keys:

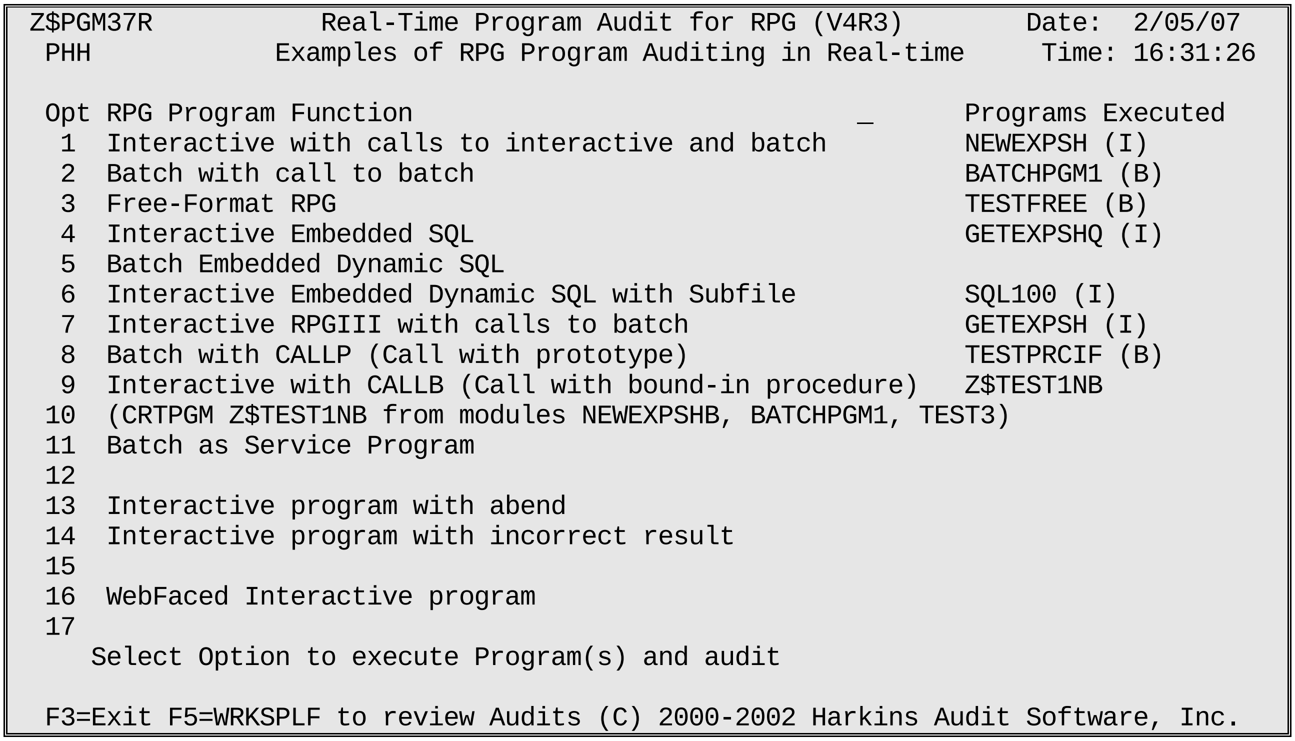
Command Key	Function
F1	Display Online Help
F3	Exit
F4	Prompt field,
F5	Refresh screen or Apply screen
F6	Auditing Options
F7	Compile Options
F8	Conditional Auditing
F9	RTPA Maintenance Menu window
F10	Submit File for Auditing
F11	Advanced Auditing
F12	Previous screen
F13	(Advanced) Select Records to Audit
F14	(Advanced) Select Variables to Audit
F15	(Advanced) Select Operations to Audit
F16	(Advanced) Select Labels and Subroutines to Audit
F17	(Advanced) Select Indicators to Audit
F18	Job History
F19	Called programs from this expanded source program
F21	Conditional Operations in program
F22	Indicators used in program
F23	Per-Audit Conditions in program
F24	More command keys (toggle)

Appendix

C

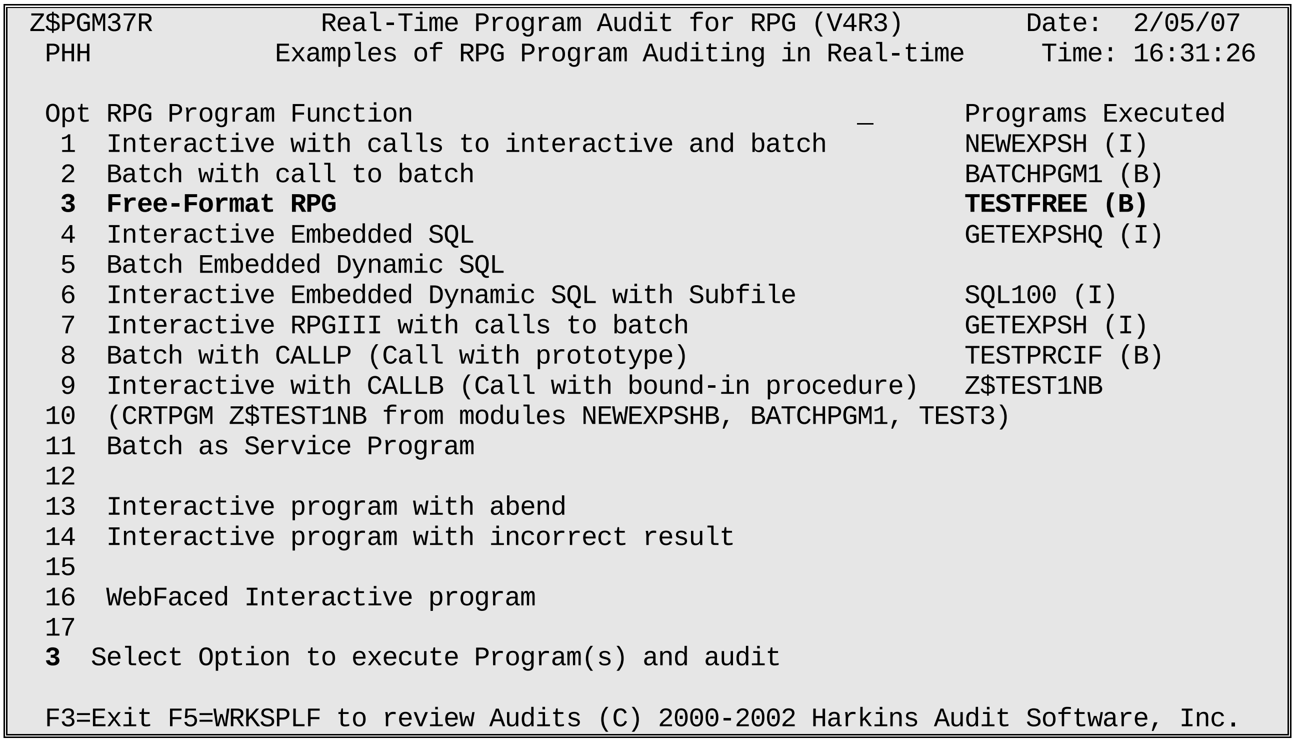
Appendix C: **RTPADEMO Menu of RPG Auditing Examples**

The RTPADEMO command displays a Menu of RTPA for RPG examples of audit enabled programs that may be selected to produce RTPA audit output in real-time as the programs execute.



```
Z$PGM37R          Real-Time Program Audit for RPG (V4R3)       Date:  2/05/07
 PHH           Examples of RPG Program Auditing in Real-time     Time: 16:31:26

 Opt RPG Program Function                                 _      Programs Executed
  1  Interactive with calls to interactive and batch            NEWEXPSH (I)
  2  Batch with call to batch                                   BATCHPGM1 (B)
  3  Free-Format RPG                                            TESTFREE (B)
  4  Interactive Embedded SQL                                   GETEXPSHQ (I)
  5  Batch Embedded Dynamic SQL
  6  Interactive Embedded Dynamic SQL with Subfile              SQL100 (I)
  7  Interactive RPGIII with calls to batch                     GETEXPSH (I)
  8  Batch with CALLP (Call with prototype)                     TESTPRCIF (B)
  9  Interactive with CALLB (Call with bound-in procedure)      Z$TEST1NB
 10  (CRTPGM Z$TEST1NB from modules NEWEXPSHB, BATCHPGM1, TEST3)
 11  Batch as Service Program
 12
 13  Interactive program with abend
 14  Interactive program with incorrect result
 15
 16  WebFaced Interactive program
 17
    Select Option to execute Program(s) and audit

 F3=Exit F5=WRKSPLF to review Audits (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure C.1 RTPADEMO command to display Menu of RTPA for RPG program examples



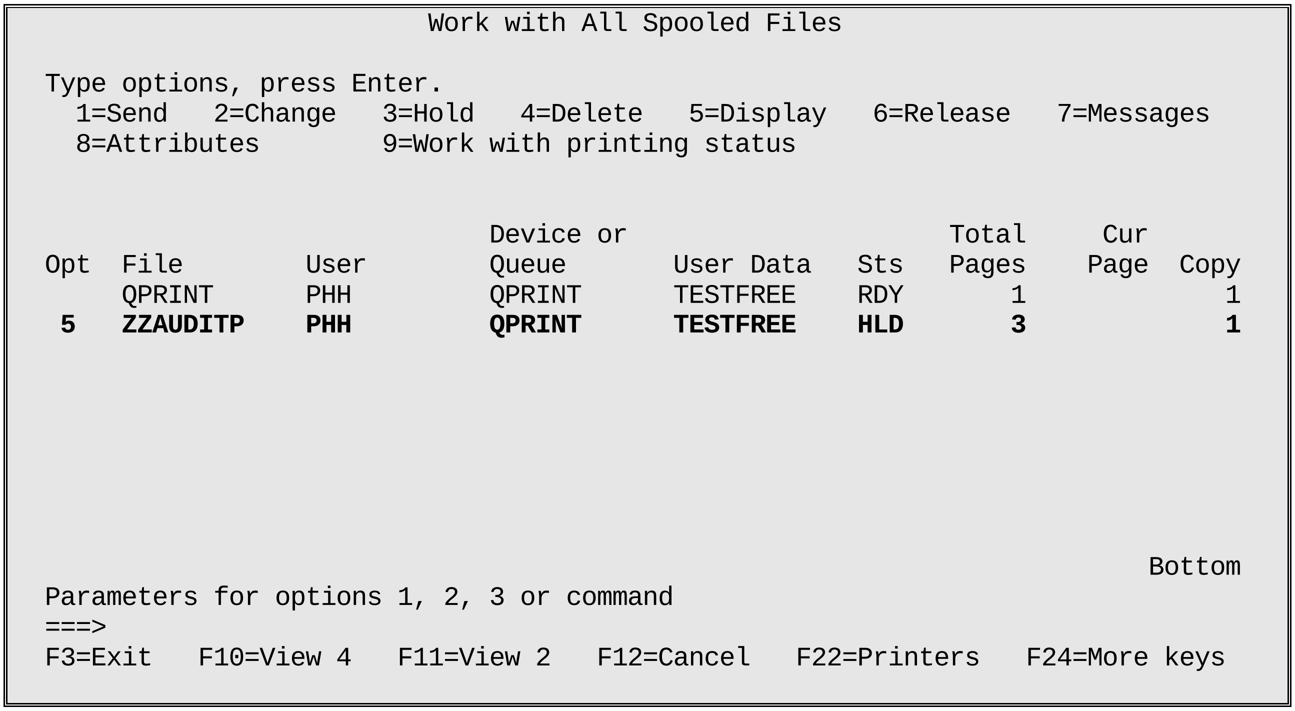
```
 Z$PGM37R          Real-Time Program Audit for RPG (V4R3)       Date:  2/05/07
  PHH           Examples of RPG Program Auditing in Real-time     Time: 16:31:26

  Opt RPG Program Function                             _      Programs Executed
   1  Interactive with calls to interactive and batch         NEWEXPSH (I)
   2  Batch with call to batch                                BATCHPGM1 (B)
   3  Free-Format RPG                                         TESTFREE (B)
   4  Interactive Embedded SQL                                GETEXPSHQ (I)
   5  Batch Embedded Dynamic SQL
   6  Interactive Embedded Dynamic SQL with Subfile           SQL100 (I)
   7  Interactive RPGIII with calls to batch                  GETEXPSH (I)
   8  Batch with CALLP (Call with prototype)                  TESTPRCIF (B)
   9  Interactive with CALLB (Call with bound-in procedure)   Z$TEST1NB
  10  (CRTPGM Z$TEST1NB from modules NEWEXPSHB, BATCHPGM1, TEST3)
  11  Batch as Service Program
  12
  13  Interactive program with abend
  14  Interactive program with incorrect result
  15
  16  WebFaced Interactive program
  17
  3  Select Option to execute Program(s) and audit

  F3=Exit F5=WRKSPLF to review Audits (C) 2000-2002 Harkins Audit Software, Inc.
```

Figure C.2 Select Option 3 to execute TESTFREE expanded program and produce ZZAUDITP file

Press F5 to Review Audit Output

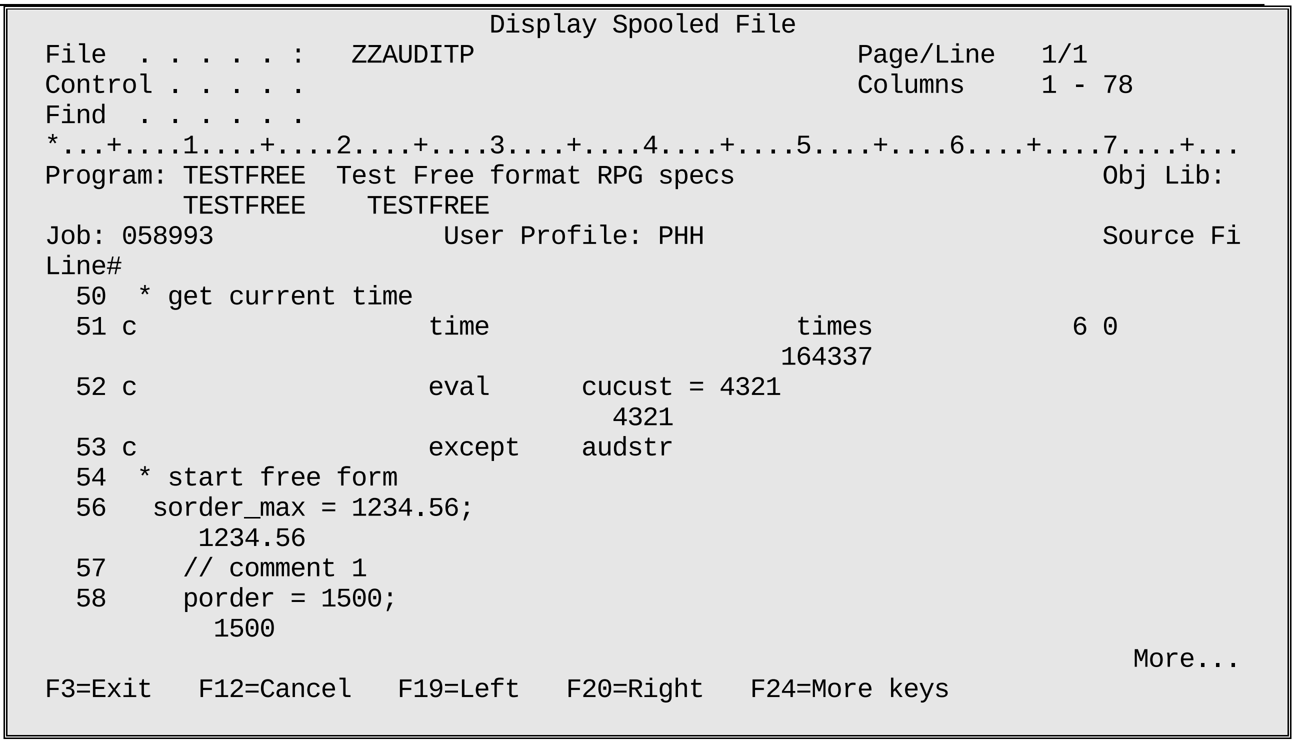
The message **Press F5 to Review Audit Output** is displayed at the bottom of the screen, indicating that the batch TESTFREE program has completed execution.

Command key 5 may be pressed to display the WRKSPLF screen to review the RTPA audit output of the TESTFREE program execution.



```
                          Work with All Spooled Files

 Type options, press Enter.
   1=Send   2=Change   3=Hold   4=Delete   5=Display   6=Release   7=Messages
   8=Attributes        9=Work with printing status


                              Device or                   Total     Cur
 Opt  File        User        Queue       User Data   Sts   Pages    Page  Copy
      QPRINT      PHH         QPRINT      TESTFREE    RDY       1             1
  5   ZZAUDITP    PHH         QPRINT      TESTFREE    HLD       3             1








                                                                         Bottom
 Parameters for options 1, 2, 3 or command
 ===>
 F3=Exit   F10=View 4   F11=View 2   F12=Cancel   F22=Printers   F24=More keys
```

Figure C.3 User WRKSPLF file with ZZAUDITP output of program TESTFREE



```
                              Display Spooled File
 File  . . . . . :   ZZAUDITP                        Page/Line   1/1
 Control . . . . .                                   Columns     1 - 78
 Find  . . . . . .
 *...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
 Program: TESTFREE  Test Free format RPG specs                       Obj Lib:
          TESTFREE    TESTFREE
 Job: 058993               User Profile: PHH                         Source Fi
 Line#
   50  * get current time
   51 c                  time                    times            6 0
                                                164337
   52 c                  eval      cucust = 4321
                                     4321
   53 c                  except    audstr
   54  * start free form
   56   sorder_max = 1234.56;
          1234.56
   57     // comment 1
   58     porder = 1500;
           1500
                                                                        More...
 F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Figure C.4 RTPA audit output ZZAUDITP file of TESTFREE for review
(Only the leftmost 80 characters of the ZZAUDITP audit file are displayed)
The complete ZZAUDITP audit file output for TESTFREE is in Appendix E.

Appendix

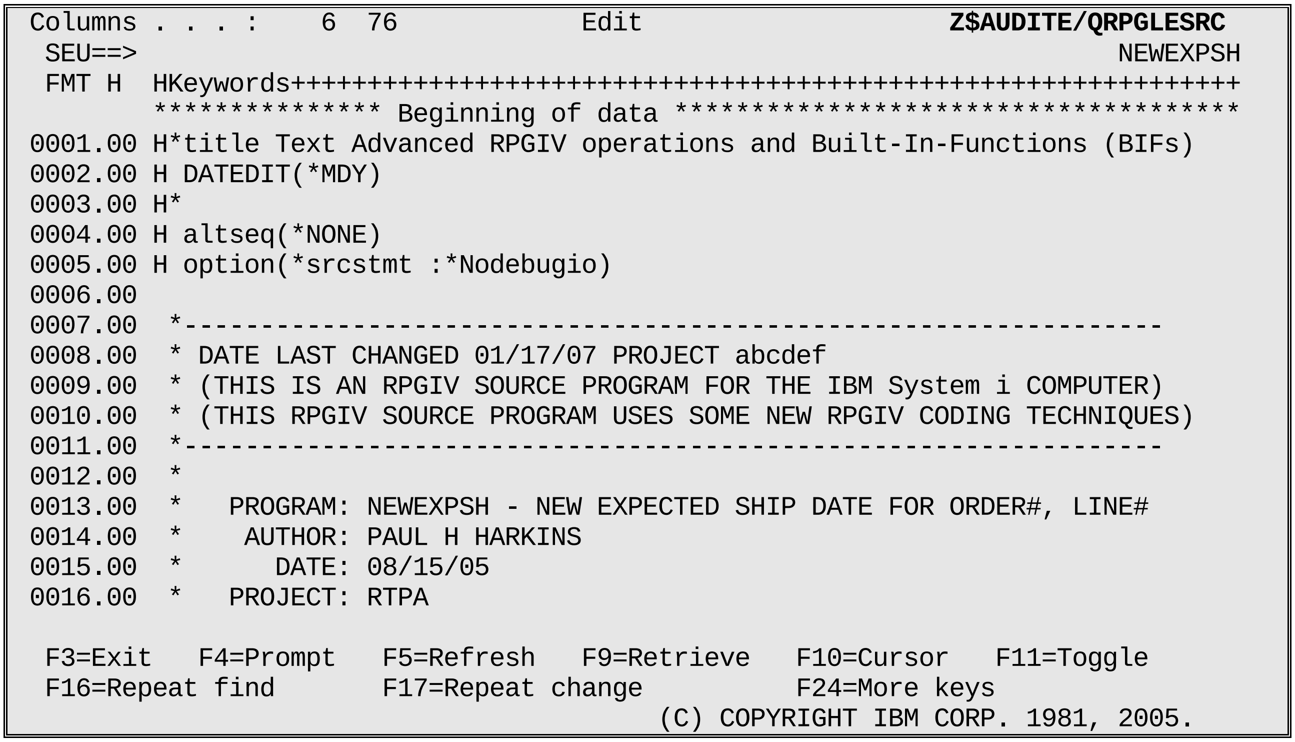
D

Appendix D: User Profile and Job Description for RTPA

Changing the RTPA Expanded source program in library Z$AUDITE and recompiling it

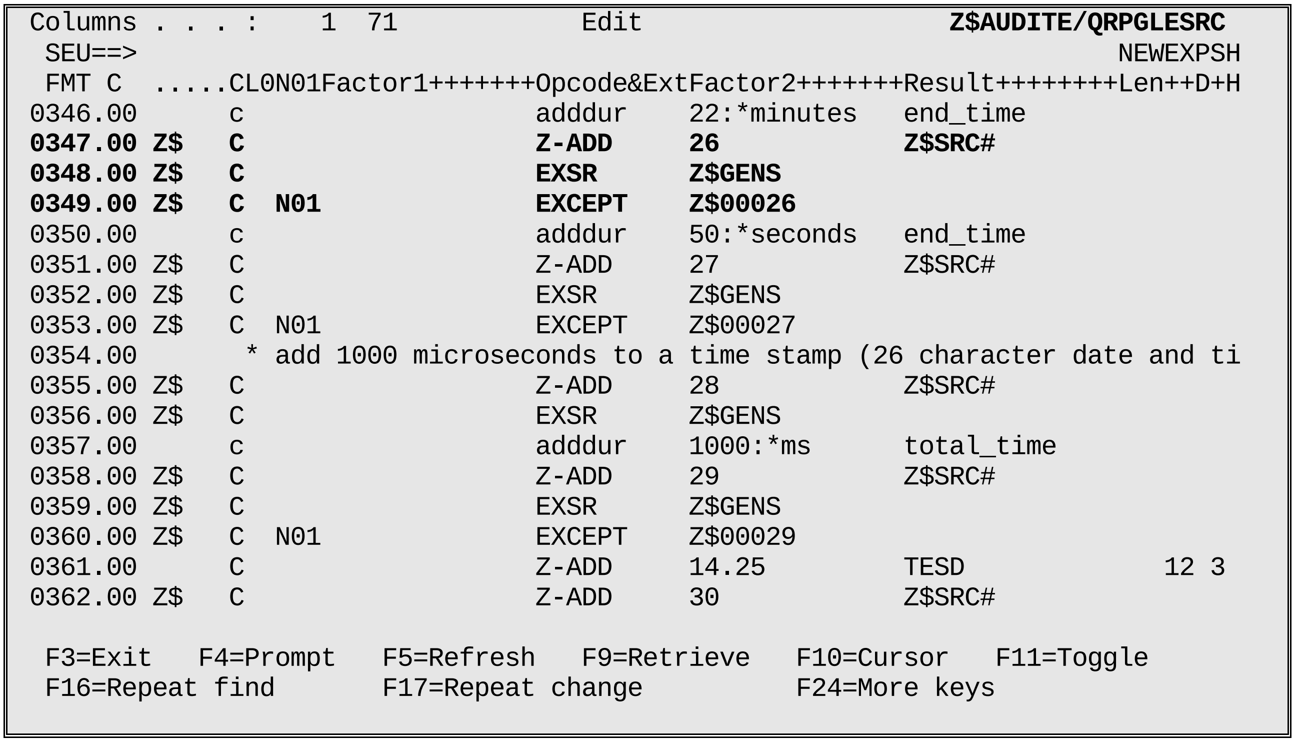
The expanded RPG source program is in library Z$AUDITE. This RPG source program contains the Z$ audit statements that produce the audit report in the printer file Z$DUDITP when the expanded object program is executed. This expanded source program in Z$AUDIT may be modified as desired and the compiled as a normal RPG program (with PDM and compiled with option 14 of PDM). RPGIII expanded source programs are in file QRPGSRC in library Z$AUDITE. RPGIV expanded source programs are in file QRPGLESRC in library Z$AUDITE.

The expanded RPG object program may be placed in any library, and executes without the need for either the Z$AUDIT or Z$AUDITE libraries in the execution library list (unless the expanded object is placed in library Z$AUDITE).



```
 Columns . . . :    6  76           Edit                       Z$AUDITE/QRPGLESRC
  SEU==>                                                                  NEWEXPSH
  FMT H  HKeywords++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
         *************** Beginning of data *************************************
 0001.00 H*title Text Advanced RPGIV operations and Built-In-Functions (BIFs)
 0002.00 H DATEDIT(*MDY)
 0003.00 H*
 0004.00 H altseq(*NONE)
 0005.00 H option(*srcstmt :*Nodebugio)
 0006.00
 0007.00  *-------------------------------------------------------------------
 0008.00  * DATE LAST CHANGED 01/17/07 PROJECT abcdef
 0009.00  * (THIS IS AN RPGIV SOURCE PROGRAM FOR THE IBM System i COMPUTER)
 0010.00  * (THIS RPGIV SOURCE PROGRAM USES SOME NEW RPGIV CODING TECHNIQUES)
 0011.00  *-------------------------------------------------------------------
 0012.00  *
 0013.00  *   PROGRAM: NEWEXPSH - NEW EXPECTED SHIP DATE FOR ORDER#, LINE#
 0014.00  *    AUTHOR: PAUL H HARKINS
 0015.00  *      DATE: 08/15/05
 0016.00  *   PROJECT: RTPA

  F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
  F16=Repeat find       F17=Repeat change          F24=More keys
                                    (C) COPYRIGHT IBM CORP. 1981, 2005.
```

Figure D.1 RTPA expanded source with Z$ Audit statements is in library Z$AUDITE
Z$AUDITE/QRGLESRC



```
 Columns . . . :    1  71           Edit                       Z$AUDITE/QRPGLESRC
  SEU==>                                                                  NEWEXPSH
  FMT C  .....CL0N01Factor1+++++++Opcode&ExtFactor2+++++++Result++++++++Len++D+H
 0346.00      c                   adddur    22:*minutes   end_time
 0347.00 Z$   C                   Z-ADD     26            Z$SRC#
 0348.00 Z$   C                   EXSR      Z$GENS
 0349.00 Z$   C  N01              EXCEPT    Z$00026
 0350.00      c                   adddur    50:*seconds   end_time
 0351.00 Z$   C                   Z-ADD     27            Z$SRC#
 0352.00 Z$   C                   EXSR      Z$GENS
 0353.00 Z$   C  N01              EXCEPT    Z$00027
 0354.00       * add 1000 microseconds to a time stamp (26 character date and ti
 0355.00 Z$   C                   Z-ADD     28            Z$SRC#
 0356.00 Z$   C                   EXSR      Z$GENS
 0357.00      c                   adddur    1000:*ms      total_time
 0358.00 Z$   C                   Z-ADD     29            Z$SRC#
 0359.00 Z$   C                   EXSR      Z$GENS
 0360.00 Z$   C  N01              EXCEPT    Z$00029
 0361.00      C                   Z-ADD     14.25         TESD             12 3
 0362.00 Z$   C                   Z-ADD     30            Z$SRC#

  F3=Exit   F4=Prompt   F5=Refresh   F9=Retrieve   F10=Cursor   F11=Toggle
  F16=Repeat find       F17=Repeat change          F24=More keys
```

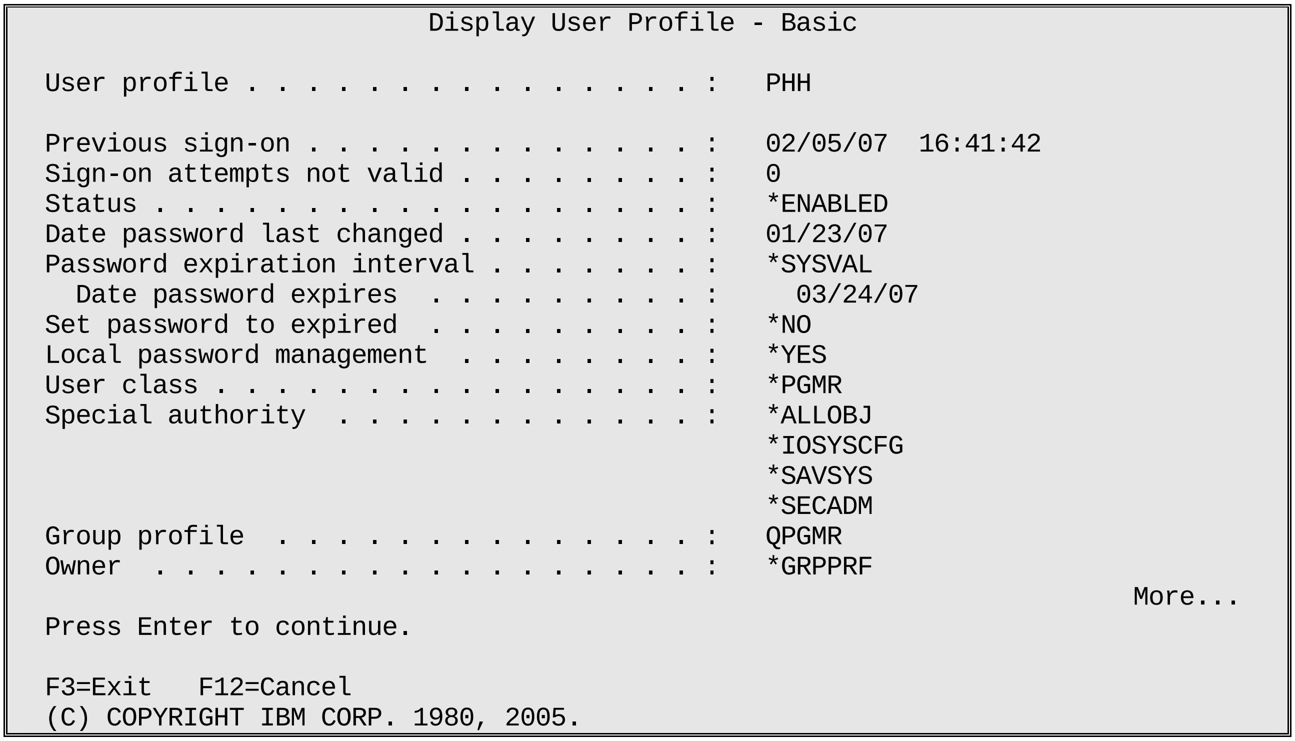
Figure D.2 RTPA expanded source Z$ Audit statements may be deleted or changed

RTPA Z$ executable Audit statements actually produce the audit output in the audit file ZZAUDITP. These Z$ audit statements may be commented or changed, if needed, and the source program

NEWEXPSH may be compiled from file QRPGLESRC in library Z$AUDITE to produce an expanded object program for auditing.

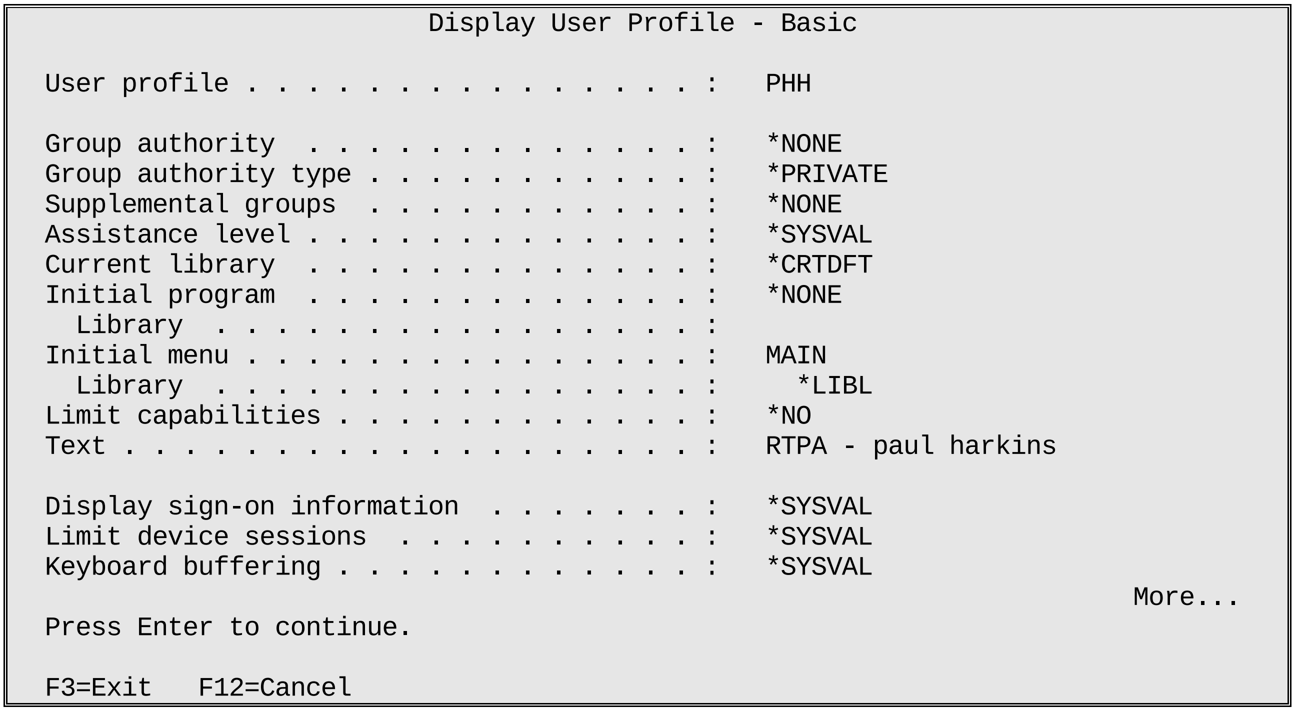
Typical Programmer Profile for RTPA expansion (PHH)

This sample Programmer User Profile uses the job description RTPA in QGPL, which contains a library list including the two required RTPA libraries Z$AUDIT and Z$AUDITE for expanding RPG input source programs.

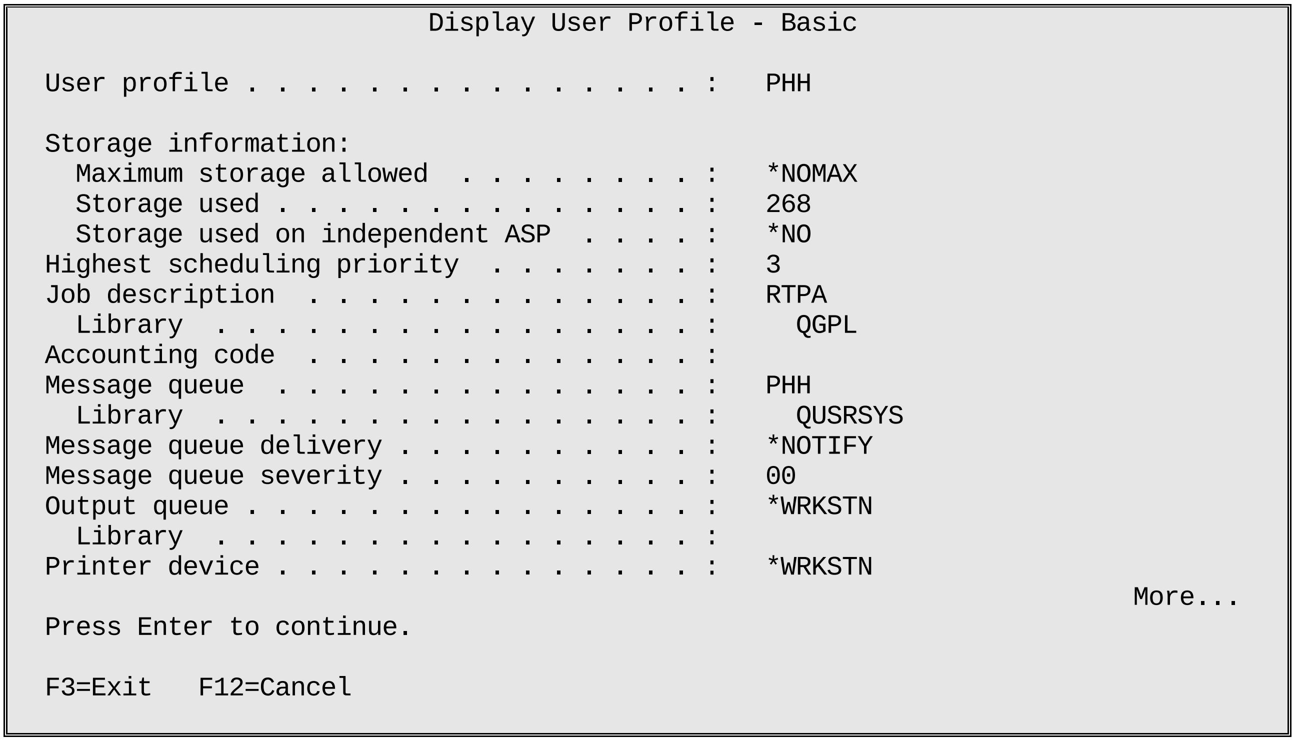


```
                           Display User Profile - Basic

  User profile . . . . . . . . . . . . . . . :   PHH

  Previous sign-on . . . . . . . . . . . . . :   02/05/07  16:41:42
  Sign-on attempts not valid . . . . . . . . :   0
  Status . . . . . . . . . . . . . . . . . . :   *ENABLED
  Date password last changed . . . . . . . . :   01/23/07
  Password expiration interval . . . . . . . :   *SYSVAL
    Date password expires  . . . . . . . . . :     03/24/07
  Set password to expired  . . . . . . . . . :   *NO
  Local password management  . . . . . . . . :   *YES
  User class . . . . . . . . . . . . . . . . :   *PGMR
  Special authority  . . . . . . . . . . . . :   *ALLOBJ
                                                 *IOSYSCFG
                                                 *SAVSYS
                                                 *SECADM
  Group profile  . . . . . . . . . . . . . . :   QPGMR
  Owner  . . . . . . . . . . . . . . . . . . :   *GRPPRF
                                                                         More...
  Press Enter to continue.

  F3=Exit   F12=Cancel
  (C) COPYRIGHT IBM CORP. 1980, 2005.
```

Figure D.3 Sample User Programmer Profile PHH

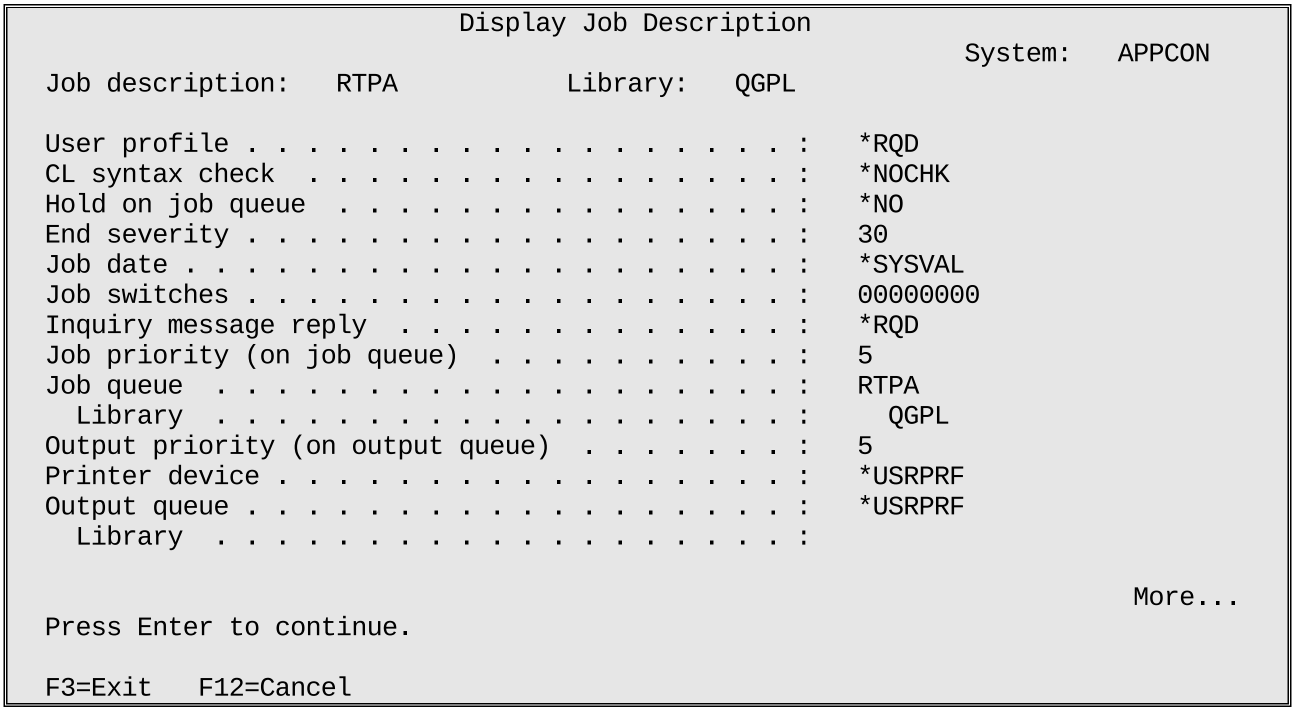


```
                         Display User Profile - Basic

User profile . . . . . . . . . . . . . . . :   PHH

Group authority  . . . . . . . . . . . . . :   *NONE
Group authority type . . . . . . . . . . . :   *PRIVATE
Supplemental groups  . . . . . . . . . . . :   *NONE
Assistance level . . . . . . . . . . . . . :   *SYSVAL
Current library  . . . . . . . . . . . . . :   *CRTDFT
Initial program  . . . . . . . . . . . . . :   *NONE
  Library  . . . . . . . . . . . . . . . . :
Initial menu . . . . . . . . . . . . . . . :   MAIN
  Library  . . . . . . . . . . . . . . . . :     *LIBL
Limit capabilities . . . . . . . . . . . . :   *NO
Text . . . . . . . . . . . . . . . . . . . :   RTPA - paul harkins

Display sign-on information  . . . . . . . :   *SYSVAL
Limit device sessions  . . . . . . . . . . :   *SYSVAL
Keyboard buffering . . . . . . . . . . . . :   *SYSVAL
                                                                       More...
Press Enter to continue.

F3=Exit   F12=Cancel
```



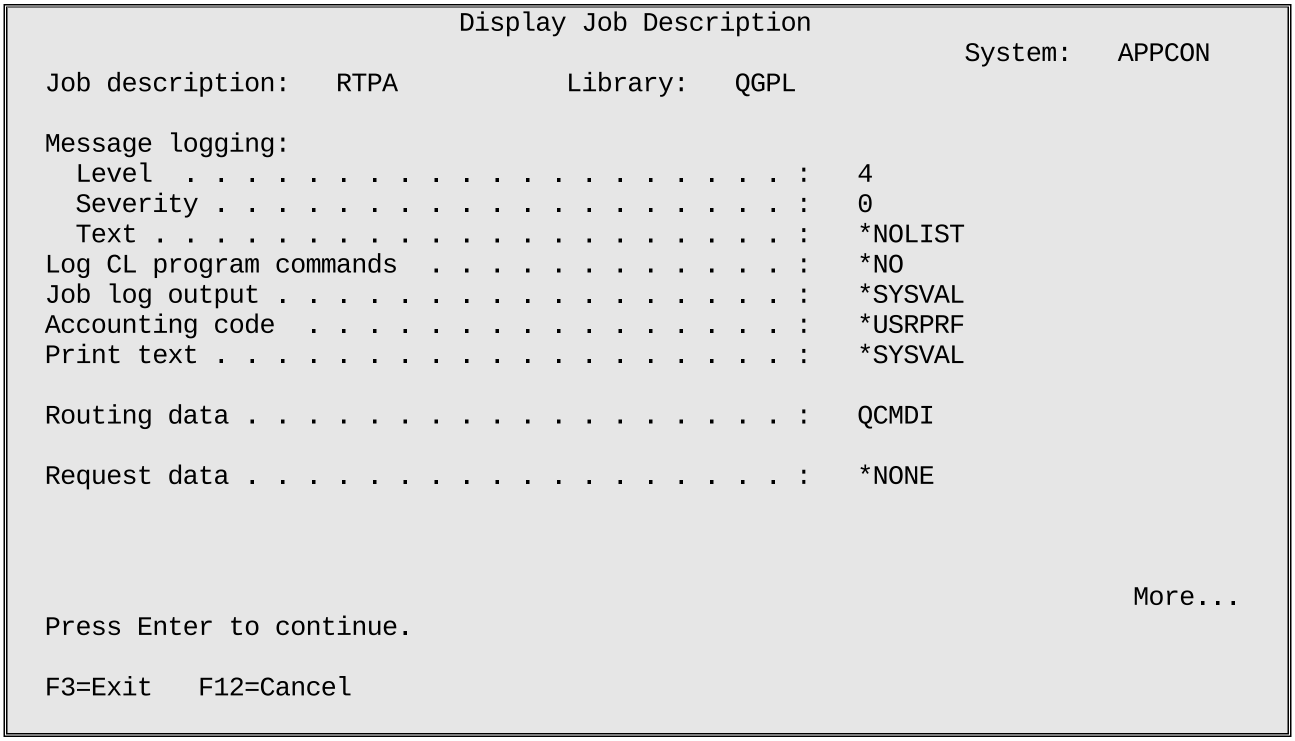
```
                         Display User Profile - Basic

User profile . . . . . . . . . . . . . . . :   PHH

Storage information:
  Maximum storage allowed  . . . . . . . . :   *NOMAX
  Storage used . . . . . . . . . . . . . . :   268
  Storage used on independent ASP  . . . . :   *NO
Highest scheduling priority  . . . . . . . :   3
Job description  . . . . . . . . . . . . . :   RTPA
  Library  . . . . . . . . . . . . . . . . :     QGPL
Accounting code  . . . . . . . . . . . . . :
Message queue  . . . . . . . . . . . . . . :   PHH
  Library  . . . . . . . . . . . . . . . . :     QUSRSYS
Message queue delivery . . . . . . . . . . :   *NOTIFY
Message queue severity . . . . . . . . . . :   00
Output queue . . . . . . . . . . . . . . . :   *WRKSTN
  Library  . . . . . . . . . . . . . . . . :
Printer device . . . . . . . . . . . . . . :   *WRKSTN
                                                                       More...
Press Enter to continue.

F3=Exit   F12=Cancel
```

RTPA Job Description in Library QGPL

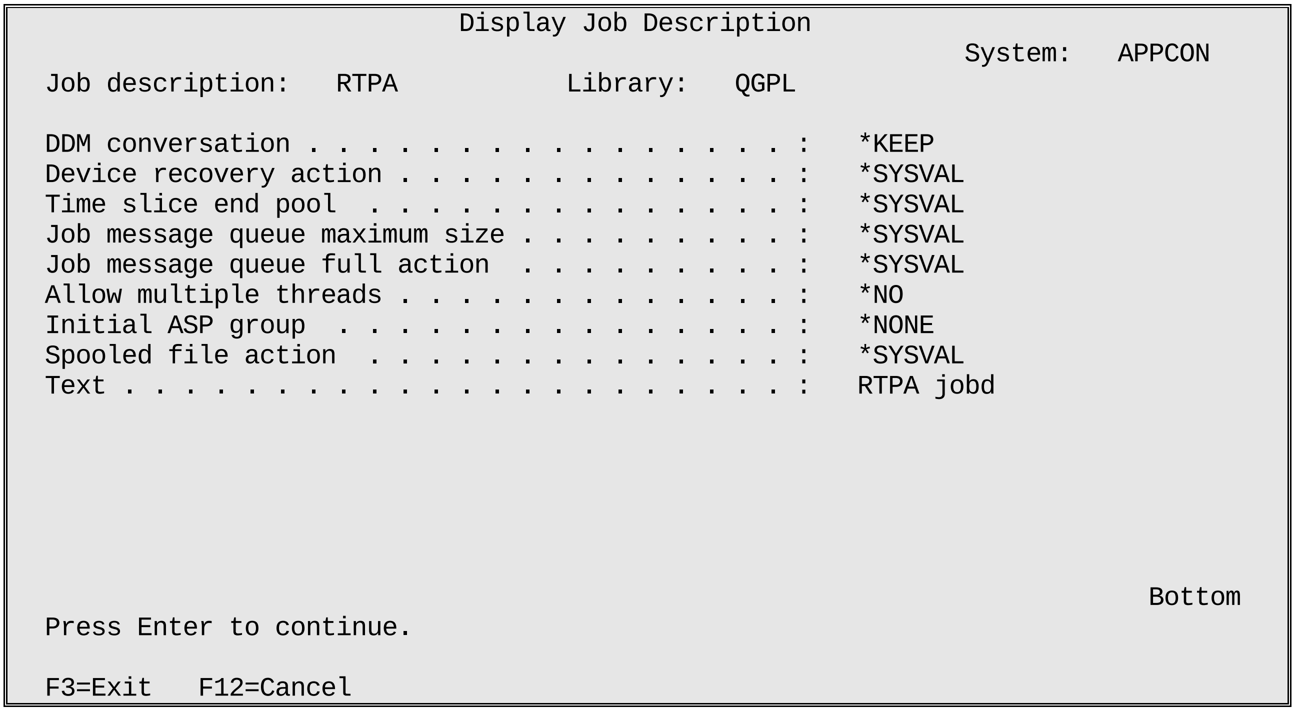


```
                           Display Job Description
                                                            System:   APPCON
Job description:   RTPA           Library:   QGPL

User profile . . . . . . . . . . . . . . . . . . :   *RQD
CL syntax check  . . . . . . . . . . . . . . . . :   *NOCHK
Hold on job queue  . . . . . . . . . . . . . . . :   *NO
End severity . . . . . . . . . . . . . . . . . . :   30
Job date . . . . . . . . . . . . . . . . . . . . :   *SYSVAL
Job switches . . . . . . . . . . . . . . . . . . :   00000000
Inquiry message reply  . . . . . . . . . . . . . :   *RQD
Job priority (on job queue)  . . . . . . . . . . :   5
Job queue  . . . . . . . . . . . . . . . . . . . :   RTPA
  Library  . . . . . . . . . . . . . . . . . . . :     QGPL
Output priority (on output queue)  . . . . . . . :   5
Printer device . . . . . . . . . . . . . . . . . :   *USRPRF
Output queue . . . . . . . . . . . . . . . . . . :   *USRPRF
  Library  . . . . . . . . . . . . . . . . . . . :

                                                                       More...
Press Enter to continue.

F3=Exit   F12=Cancel
```

Figure D.4 Sample Job Description RTPA



```
                           Display Job Description
                                                            System:   APPCON
Job description:   RTPA           Library:   QGPL

Message logging:
  Level  . . . . . . . . . . . . . . . . . . . . :   4
  Severity . . . . . . . . . . . . . . . . . . . :   0
  Text . . . . . . . . . . . . . . . . . . . . . :   *NOLIST
Log CL program commands  . . . . . . . . . . . . :   *NO
Job log output . . . . . . . . . . . . . . . . . :   *SYSVAL
Accounting code  . . . . . . . . . . . . . . . . :   *USRPRF
Print text . . . . . . . . . . . . . . . . . . . :   *SYSVAL

Routing data . . . . . . . . . . . . . . . . . . :   QCMDI

Request data . . . . . . . . . . . . . . . . . . :   *NONE



                                                                       More...
Press Enter to continue.

F3=Exit   F12=Cancel
```



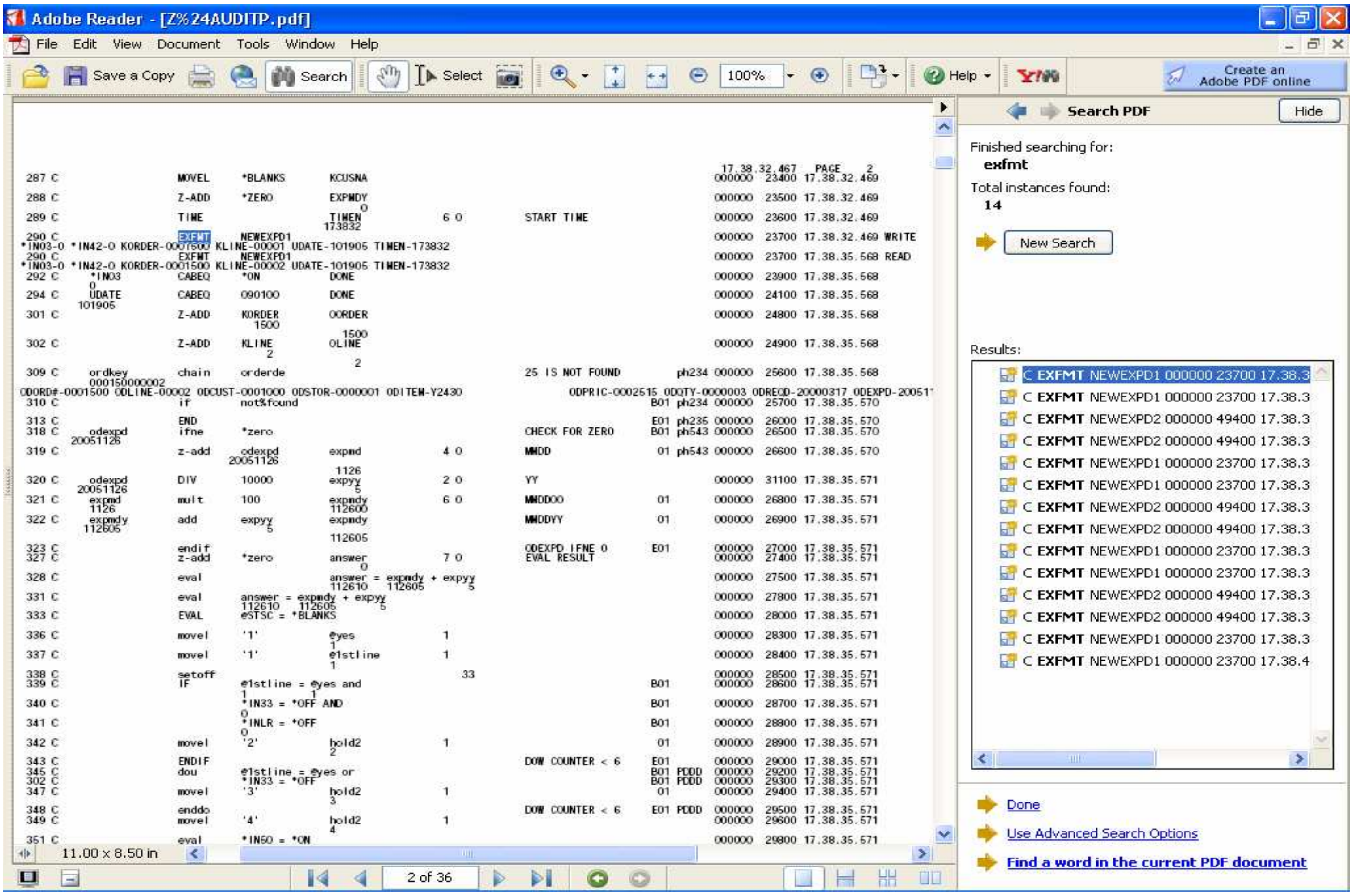
```
                           Display Job Description
                                                          System:   APPCON
 Job description:   RTPA           Library:   QGPL

 DDM conversation . . . . . . . . . . . . . . . . . :   *KEEP
 Device recovery action . . . . . . . . . . . . . . :   *SYSVAL
 Time slice end pool  . . . . . . . . . . . . . . . :   *SYSVAL
 Job message queue maximum size . . . . . . . . . . :   *SYSVAL
 Job message queue full action  . . . . . . . . . . :   *SYSVAL
 Allow multiple threads . . . . . . . . . . . . . . :   *NO
 Initial ASP group  . . . . . . . . . . . . . . . . :   *NONE
 Spooled file action  . . . . . . . . . . . . . . . :   *SYSVAL
 Text . . . . . . . . . . . . . . . . . . . . . . . :   RTPA jobd






                                                                            Bottom
 Press Enter to continue.

 F3=Exit   F12=Cancel
```

Appendix

E

Appendix E: **RTPA Audit Output Examples**

Examples of Input RPG programs and RTPA Audit Output



Adobe Reader - [Z%24AUDITP.pdf]
File Edit View Document Tools Window Help
Save a Copy
Search
Select
100%
Help
Create an Adobe PDF online
11.00 x 8.50 in
2 of 36

```
                                                                                    17.38.32.467   PAGE    2
 287 C                  MOVEL     *BLANKS       KCUSNA                                        000000  23400 17.38.32.469

 288 C                  Z-ADD     *ZERO         EXPMDY                                        000000  23500 17.38.32.469
                                                     0
 289 C                  TIME                    TIMEN           6 0    START TIME             000000  23600 17.38.32.469
                                               173832
 290 C                  EXFMT     NEWEXPD1                                                    000000  23700 17.38.32.469 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-101905 TIMEN-173832
 290 C                  EXFMT     NEWEXPD1                                                    000000  23700 17.38.35.568 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-101905 TIMEN-173832
 292 C     *IN03        CABEQ     *ON           DONE                                          000000  23900 17.38.35.568
           0
 294 C     UDATE        CABEQ     090100        DONE                                          000000  24100 17.38.35.568
         101905
 301 C                  Z-ADD     KORDER        OORDER                                        000000  24800 17.38.35.568
                                    1500
                                                  1500
 302 C                  Z-ADD     KLINE         OLINE                                         000000  24900 17.38.35.568
                                       2
                                                     2
 309 C     ordkey       chain     orderde                      25 IS NOT FOUND         ph234 000000  25600 17.38.35.568
           000150000002
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430          ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317 ODEXPD-20051
 310 C                  if        not%found                                          B01 ph234 000000  25700 17.38.35.570

 313 C                  END                                                          E01 ph235 000000  26000 17.38.35.570
 318 C     odexpd       ifne      *zero                        CHECK FOR ZERO        B01 ph543 000000  26500 17.38.35.570
        20051126
 319 C                  z-add     odexpd        expmd           4 0    MMDD            01 ph543 000000  26600 17.38.35.570
                                20051126
                                                  1126
 320 C     odexpd       DIV       10000         expyy           2 0    YY                      000000  31100 17.38.35.571
         20051126                                   5
 321 C     expmd        mult      100           expmdy          6 0    MMDDOO          01      000000  26800 17.38.35.571
           1126                                 112600
 322 C     expmdy       add       expyy         expmdy                 MMDDYY          01      000000  26900 17.38.35.571
          112605                      5
                                                112605
 323 C                  endif                                          ODEXPD IFNE 0   E01     000000  27000 17.38.35.571
 327 C                  z-add     *zero         answer          7 0    EVAL RESULT             000000  27400 17.38.35.571
                                                     0
 328 C                  eval                    answer = expmdy + expyy                        000000  27500 17.38.35.571
                                                112610   112605       5
 331 C                  eval      answer = expmdy + expyy                                      000000  27800 17.38.35.571
                                  112610   112605       5
 333 C                  EVAL      @STSC = *BLANKS                                              000000  28000 17.38.35.571

 336 C                  movel     '1'           @yes            1                              000000  28300 17.38.35.571
                                                1
 337 C                  movel     '1'           @1stline        1                              000000  28400 17.38.35.571
                                                1
 338 C                  setoff                                   33                            000000  28500 17.38.35.571
 339 C                  IF        @1stline = @yes and                                  B01     000000  28600 17.38.35.571
                                  1          1
 340 C                            *IN33 = *OFF AND                                     B01     000000  28700 17.38.35.571
                                  0
 341 C                            *INLR = *OFF                                         B01     000000  28800 17.38.35.571
                                  0
 342 C                  movel     '2'           hold2           1                       01     000000  28900 17.38.35.571
                                                2
 343 C                  ENDIF                                          DOW COUNTER < 6  E01     000000  29000 17.38.35.571
 345 C                  dou       @1stline = @yes or                                   B01 PDDD 000000  29200 17.38.35.571
 302 C                            *IN33 = *OFF                                         B01 PDDD 000000  29300 17.38.35.571
 347 C                  movel     '3'           hold2           1                       01     000000  29400 17.38.35.571
                                                3
 348 C                  enddo                                          DOW COUNTER < 6  E01 PDDD 000000  29500 17.38.35.571
 349 C                  movel     '4'           hold2           1                              000000  29600 17.38.35.571
                                                4
 351 C                  eval      *IN50 = *ON                                                  000000  29800 17.38.35.571
```

Search PDF
Hide
New Search
Done
Use Advanced Search Options
Find a word in the current PDF document

Finished searching for:
exfmt

Total instances found:
14

Results:

```
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD2 000000 49400 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.3
C EXFMT NEWEXPD1 000000 23700 17.38.4
```

Input Source RPGLE program TESTEVALCO   Test EVAL_CORR OP CODE V5R4

```
Columns . . . :    6  76           Edit                     Z$AUDIT/QRPGLESRC
 SEU==>                                                             TESTEVALCO
 FMT D  DName+++++++++++ETDsFrom+++To/L+++IDc.Keywords+++++++++++++++++++++++++
        *************** Beginning of data ************************************
0001.00 d DSa             ds                 Qualified
0002.00 d aorder                        8  0 inz(0)
0003.00 d xorder                        9  2 inz(0)
0004.00 d iorder                        7  3 inz(0)
0005.00 d rorder                        9  3 inz(0)
0006.00  *
0007.00 d DSb             ds                 Qualified
0008.00 d border                        8  0 inz(0)
0009.00 d xorder                        9  2 inz(0)
0010.00 d iorder                        7  3 inz(0)
0011.00 d rorder                        9  3 inz(0)
0011.01 c                   Time                    Timen             6 0
0011.02  * audit the RPGLE V5R4 new Op code eval-corr
0011.03  *  DSa is a qualified data structure
0011.04  *DSa             ds                 Qualified
0011.05  *aorder                        8  0 inz(0)
0011.06  *xorder                        9  2 inz(0)
0011.07  *iorder                        7  3 inz(0)
0011.08  *rorder                        6  3 inz(0)
0011.09  * DSa DS fields xorder, iorder, and rorder correspond with DSb DS
0011.10  *  DSb is a qualified data structure
0011.11  *DSb             ds                 Qualified
0011.12  *border                        8  0 inz(0)
0011.13  *xorder                        9  2 inz(0)
0011.14  *iorder                        7  3 inz(0)
0011.15  *rorder                        6  3 inz(0)
0011.16  *  fill fields in qualifed DS DSa fields
0012.00 c                   Eval      DSa.aorder = 1500
0013.00 c                   Eval      DSa.xorder = 1234567.89
0013.01 c                   Eval      DSa.iorder = 7654.321
0013.02 c                   Eval      DSa.rorder = 123.456
0017.00  /free
0017.01   // in free form, fill qualifed DS DSb field border
0018.00         DSb.border = DSa.aorder + 13.45  +
0019.00                      26.2 + DSa.iorder;
0021.00  /end-free
0022.00  * V5R4 op code eval-corr (evaluate corresponding)
0023.00 c                   eval-corr DSb = DSa
0024.00  * qualified DS names (DSb.)
0025.00 c                   z-add     DSb.border    work80            8 0
0025.01 c                   z-add     DSb.xorder    work92            9 2
0025.02 c                   z-add     DSb.iorder    work73            7 3
0026.00 c                   z-add     DSb.rorder    work63            6 3
0027.00  * exit program
0027.01 c                   Eval      *inlr = *on
0028.00 c                   Return
        ****************** End of data ****************************************
```

RTPA Audit Output for RPGLE program TESTEVALCO

```
Program: TESTEVALCOTest Eval-corr  corresponding                    Obj Lib: Z$AUDITE    Initiated: 1/25/07 18.27.06.818   PAGE
          TESTEVALCO  TESTEVALCO
 Job: 054067              User Profile: PHH                     Source File/Library:  QRPGLESRC Z$AUDIT
 Line#                                                                             Do# SrcId ChgDat  Seq#   Time
  12 c                   Time                    Timen             6 0    Get the current          070124    1101 18.27.06.818
                                                 182706
  13  * audit the RPGLE V5R4 new Op code eval-corr                                         070125   1102 18.27.06.818
  14  *  DSa is a qualified data structure                                                 070125   1103 18.27.06.818
  15  *DSa             ds                 Qualified                                        070125   1104 18.27.06.818
  16  *aorder                        8  0 inz(0)                                           070125   1105 18.27.06.818
  17  *xorder                        9  2 inz(0)                                           070125   1106 18.27.06.818
  18  *iorder                         7  3 inz(0)                                            070125   1107 18.27.06.818
  19  *rorder                         6  3 inz(0)                                            070125   1108 18.27.06.818
  20  * DSa DS fields xorder, iorder, and rorder correspond with DSb DS                      070125   1109 18.27.06.818
  21  *  DSb is a qualified data structure                                                   070125   1110 18.27.06.818
  22  *DSb             ds                  Qualified                                         070125   1111 18.27.06.818
  23  *border                         8  0 inz(0)                                            070125   1112 18.27.06.818
  24  *xorder                         9  2 inz(0)                                            070125   1113 18.27.06.818
  25  *iorder                         7  3 inz(0)                                            070125   1114 18.27.06.818
  26  *rorder                         6  3 inz(0)                                            070125   1115 18.27.06.818
  27  *  fill fields in qualifed DS dsa fields                                               070125   1116 18.27.06.818
  28 c                   Eval      DSa.aorder = 1500
                                         1500                                                070125   1200 18.27.06.818
  29 c                   Eval      DSa.xorder = 1234567.89                                   070125   1300 18.27.06.819
                                   1234567.89
  30 c                   Eval      DSa.iorder = 7654.321                                      070125   1301 18.27.06.819
                                     7654.321
  31 c                   Eval      DSa.rorder = 123.456                                      070125   1302 18.27.06.819
                                      123.456
  33   // in free form, fill qualifed DS DSb field border                                    070125   1701 18.27.06.819
  34         DSb.border = DSa.aorder + 13.45  +                                              070125   1800 18.27.06.819
                   9193         1500
  35                      26.2 + DSa.iorder;                                                 070125   1900 18.27.06.819
                                   7654.321
  37  * V5R4 op code eval-corr (evaluate corresponding)                                      070125   2200 18.27.06.819
  38 c                   eval-corr DSb = DSa                                                 070125   2300 18.27.06.819
                                   000091931234567897654321000123456
                                         000015001234567897654321000123456
  39  * qualified DS names (DSb.)                                                            070125   2400 18.27.06.819
  40 c                   z-add     DSb.border    work80            8 0                       070125   2500 18.27.06.819
                                         9193
                                                   9193
  41 c                   z-add     DSb.xorder    work92            9 2                       070125   2501 18.27.06.819
                                   1234567.89
                                             1234567.89
  42 c                   z-add     DSb.iorder    work73            7 3                       070125   2502 18.27.06.819
                                     7654.321
                                               7654.321
  43 c                   z-add     DSb.rorder    work63            6 3                       070125   2600 18.27.06.819
                                     123.456
                                                123.456
  44  * exit program                                                                         070125   2700 18.27.06.819
  45 c                   Eval      *inlr = *on                                               070125   2701 18.27.06.819
                                   1
  46 c                   Return                                                              070125   2800 18.27.06.819
```

Input Source RPGLE program TESTFREE

```
0001.00  * test of free format RPGIV source stmts (between /free and /end-free)
0002.00 fcustmast  uf   e           k disk
0003.00 fqprint    o    f  132        printer
0004.00 f                                     oflind(*inOB)
0005.00  *
0006.00 d Ws_Edate        s             10    inz
0007.00 d start_date      s               d   DATFMT(*ISO) inz(D'1998-12-18')
0008.00 d end_date        s               d   DATFMT(*ISO)
0009.00 d month_end       s               d   DATFMT(*ISO) inz(D'1994-10-31')
0010.00 d gotname         s             25
0011.00 d gotad1          s             25
0012.00 d gotad2          s             25
0013.00 d gotcity         s             25
0014.00 d                 ds
0015.00 d parmre                  1     70
0016.00 d porder                  1      7  0
0017.00 d xorder                  8     16  2
0018.00 d iorder                 17     24  3
0019.00 d rorder                 25     33  3
0020.00 d sorder_max             34     39  2
0021.00 d zorder_final           40     47  4
```

```
d scust                  48     54  0
d fstore                 55     61  0
d movesw2                62     63
c     custkey       klist
c                   kfld                    cucust
c                   kfld                    custor

 * get current time
c                   time                    times             6 0
c                   eval      cucust = 4321
c                   except    audstr
 * start free form
 /free
  sorder_max = 1234.56;
    // comment 1
    porder = 1500;
      // comment 2
     exsr moveit;
      iorder = 78.543;
  // complex free form statement compute xorder
        xorder = porder + 13.45  +
    // this is a continuation free form statement preceeded with +
                       26.2 + iorder;
     if porder <= 1500;
        xorder = porder + 87.43  +
                     1103.5 + iorder;
       // comment 3
     endif;
         // comment 4
            zorder_final = porder - iorder + xorder + sorder_max;
     scust=1000;
       // compound if group
     if porder >= 1500 or             +
        iorder = xorder  or          +
        xorder >= 1.1234;
        xorder = porder + (porder * 2) + 87.43  +
                     1103.5 + iorder;
       // comment 3
     endif;
     setll scust custmast;
     reade scust custmast;

     dow not %eof( custmast );
     fstore = custor;
  // output report line
     except prtfre;
     reade scust custmast;
     enddo;
     monitor;
     on-error;
     endmon;

      eval cucust = 1000;
      eval custor = 522;
       chain custkey custmast;  // chain on file name
        if %found( custmast );
         gotname = cuname;
         gotad1 = cuad1;
         gotad2 = cuad2;
         gotcity = cucity;
         except prtrec;
        update custrec;  // update
        endif;
      eval custor = 999;     // is no store 999
       chain custkey custmast;  // chain on file name
        if %found( custmast );
         gotname = cuname;
         gotad1 = cuad1;
         gotad2 = cuad2;
         gotcity = cucity;
         except prtrec;
        endif;
```

```
0094.00   // end of free formne
0095.00    /end-free
0096.00 c                  eval      rorder = iorder +98  +
0097.00 c                            13.5 + porder
0098.00 c                  eval      cucust = 1000
0099.00 c     cucust       setll     custrec
0100.00  * read by record name
0101.00 c     cucust       reade     custrec                                 84
0102.00 c     *in84        doweq     *off
0103.00 c                  except    prtrec
0104.00 c     cucust       reade     custrec                                 84
0105.00 c                  enddo
0106.00 c                  eval      *inlr=*on
0107.00  * exit program
0108.00 c                  return
0109.00  /free
0110.00   sorder_max = porder + iorder + xorder;
0111.00  /end-free
0112.00 c     moveit       begsr
0113.00 c                  eval      movesw2 = '11'
0114.00 c                  endsr
0115.00 oqprint    e           audstr         2 01
0116.00 o                                         8 'TESTFREE'
0117.00 o                      udate             18 '  /  /  '
0118.00 o                      times             28 '  :  :  '
0119.00 o                                        70 'Test Free Form RPGIV
0120.00  * audit input record
0121.00 o          e           prtrec         1
0122.00 o                      cucust        z      7
0123.00 o                      custor        z     15
0124.00 o                      cuname              42
0125.00 o                      cuad1               69
0126.00 o                      cuad2               95
0127.00 o                      cucity             121
0128.00 o                      custa              124
0129.00 o                                         132 'prtrec'
0130.00 o          e           prtfre         1
0131.00 o                      cucust        z      7
0132.00 o                      custor        z     15
0133.00 o                      cuname              42
0134.00 o                      cuad1               69
0135.00 o                      cuad2               95
0136.00 o                      fstore        z    103
0137.00 o                                         132 'prtfre'
        ****************** End of data ****************************************
-------------------------------------------------------------------------------------------------
```

RTPA for RPG Audit Output – TESTFREE Free Format RPG (Batch)

Note – This entire program execution and RTPA Audit took 0.032 ELAPSED second (End Time 11.10.13.627 minus Start Time 11.10.13.595)

```
Program: TESTFREE   Test Free format RPG specs        Obj Lib: Z$AUDITE     Initiated:  12/09/06 11.10.13.595     PAGE    1
TESTFREE    TESTFREE
Job: 026982              User Profile: PHH                          Source File/Library:  QRPGLESRC Z$AUDIT
Line#                                                                         Do# SrcId ChgDat   Seq#   Time
  50  * get current time                                                                060319   2900 11.10.13.595
  51 c                  time                    times            6 0                    030207   3000 11.10.13.595
                                               111013
  52 c                  eval      cucust = 4321                                         030207   3100 11.10.13.595
                                     4321
  53 c                  except    audstr                                                030207   3200 11.10.13.595
  54  * start free form                                                                 060319   3300 11.10.13.595
  56   sorder_max = 1234.56;                                                            020721   3500 11.10.13.595
          1234.56
  57     // comment 1                                                                   030207   3600 11.10.13.595
  58     porder = 1500;                                                                  020706   3700 11.10.13.595
           1500
  59       // comment 2                                                                  030207   3800 11.10.13.595
  60      exsr moveit;                                                                   030207   3900 11.10.13.595
  61       iorder = 78.543;                                                              020618   4000 11.10.13.595
           78.543
  62   // complex free form statement compute xorder                                     060319   4100 11.10.13.596
  63         xorder = porder + 13.45  +                                                  020721   4200 11.10.13.596
            1618.19     1500
  64     // this is a continuation free form statement preceeded with +                  060319   4300 11.10.13.596
  65                        26.2 + iorder;                                               020721   4400 11.10.13.596
                                   78.543
  66      if porder <= 1500;                                                   B01       030207   4500 11.10.13.596
               1500
  67         xorder = porder + 87.43  +                                         01       030207   4600 11.10.13.596
            2769.47     1500
  68                      1103.5 + iorder;                                      01       030207   4700 11.10.13.596
                                   78.543
  69        // comment 3                                                                 030207   4800 11.10.13.596
  70      endif;                                                               E01       030207   4900 11.10.13.596
  71          // comment 4                                                               030207   5000 11.10.13.596
  72              zorder_final = porder - iorder + xorder + sorder_max;                  020721   5100 11.10.13.596
                    5425.4870     1500
                                         78.543
                                                 2769.47      1234.56
  73      scust=1000;                                                                    030207   5200 11.10.13.597
           1000
  74        // compound if group                                                         030218   5300 11.10.13.597
  75      if porder >= 1500 or            +                                    B01       030218   5400 11.10.13.597
               1500
  75      if porder >= 1500 or            +                                    B01       030218   5400 11.10.13.597
               1500
  76         iorder = xorder  or         +                                     B01       030218   5500 11.10.13.597
             78.543
                     2769.47
  77         xorder >= 1.1234;                                                 B01       030218   5600 11.10.13.597
            2769.47
  78         xorder = porder + (porder * 2) + 87.43  +                          01       030218   5700 11.10.13.598
            5769.47     1500      1500
  79                      1103.5 + iorder;                                      01       030218   5800 11.10.13.598
                                   78.543
  80        // comment 3                                                                 030218   5900 11.10.13.598
  81      endif;                                                               E01       030218   6000 11.10.13.598
  82      setll scust custmast;                                                          030207   6100 11.10.13.598
                 1000
           0001000
  83      reade scust custmast;                                                          030207   6200 11.10.13.604
                 1000
           0001000
           0001000
CUCUST-0001000  CUSTOR-0000000  CUNAME-ABC  STORES  INC.                     CUAD1-15  CORPORATE  DRIVE                CUAD2-
CUCITY-WEST CHESTER              CUSTA-PA CUZIP-19382
CUCUST-0001000  CUSTOR-0000000  CUNAME-ABC  STORES  INC.                     CUAD1-15  CORPORATE  DRIVE                CUAD2-
CUCITY-WEST CHESTER              CUSTA-PA CUZIP-19382
CUCUST-0001000  CUSTOR-0000000  CUNAME-ABC  STORES  INC.                     CUAD1-15  CORPORATE  DRIVE                CUAD2-
CUCITY-WEST CHESTER              CUSTA-PA CUZIP-19382
CUCUST-0001000  CUSTOR-0000000  CUNAME-ABC  STORES  INC.                     CUAD1-15  CORPORATE  DRIVE                CUAD2-
CUCITY-WEST CHESTER              CUSTA-PA CUZIP-19382
  85      dow not %eof( custmast );                                            B01       030207   6400 11.10.13.605
  86      fstore = custor;                                                      01       030207   6500 11.10.13.605
               0        0
  87   // output report line                                                             060319   6600 11.10.13.605
  88      except prtfre;                                                        01       030207   6700 11.10.13.605
  89      reade scust custmast;                                                 01       030207   6800 11.10.13.610
                 1000
           0001000
CUCUST-0001000  CUSTOR-0000001  CUNAME-ABC  STORES  INC                        CUAD1-423  MONTGOMERY  AVENUE         CUAD2-
CUCITY-ARDMORE                   CUSTA-PA CUZIP-19333
  86      fstore = custor;                                                      01       030207   6500 11.10.13.610
               1        1
  87   // output report line                                                             060319   6600 11.10.13.610
```

```
                                                                                                       11.10.13.595     PAGE
2
  88      except prtfre;                                                        01       030207   6700 11.10.13.610
  89      reade scust custmast;                                                 01       030207   6800 11.10.13.610
                 1000
           0001000
CUCUST-0001000  CUSTOR-0000002  CUNAME-ABC  STORES    STORE  #2           CUAD1-554  ARCH  STREET                     CUAD2-
CUCITY-PHILADELPHIA              CUSTA-PA CUZIP-19025
  86      fstore = custor;                                                      01       030207   6500 11.10.13.610
               2        2
  87   // output report line                                                          060319   6600 11.10.13.610
  88      except prtfre;                                                        01       030207   6700 11.10.13.623
  89      reade scust custmast;                                                 01       030207   6800 11.10.13.623
                 1000
           0001000
CUCUST-0001000  CUSTOR-0000522  CUNAME-ABC  STORES    STORE  #522        CUAD1-231  70TH  STREET                      CUAD2-
CUCITY-NEW YORK                  CUSTA-NY CUZIP-10021
  86      fstore = custor;                                                      01       030207   6500 11.10.13.623
             522      522
  87   // output report line                                                          060319   6600 11.10.13.623
  88      except prtfre;                                                        01       030207   6700 11.10.13.623
  89      reade scust custmast;                                                 01       030207   6800 11.10.13.623
                 1000
           0001000
  90      enddo;                                                               E01       030207   6900 11.10.13.623
  91      monitor;                                                             B01       030207   7000 11.10.13.623
  92      on-error;                                                            X01       030207   7100 11.10.13.623
  93      endmon;                                                              E01       030207   7200 11.10.13.623
  95       eval cucust = 1000;                                                           030207   7400 11.10.13.624
                  1000
  96       eval custor = 522;                                                            030207   7500 11.10.13.624
                   522
  97        chain custkey custmast;  // chain on file name                               030207   7600 11.10.13.624
           00010000000522
CUCUST-0001000  CUSTOR-0000522  CUNAME-ABC  STORES    STORE  #522        CUAD1-231  70TH  STREET                      CUAD2-
CUCITY-NEW YORK                  CUSTA-NY CUZIP-10021
  98         if %found( custmast );                                            B01       030207   7700 11.10.13.624
  99          gotname = cuname;                                                 01       030207   7800 11.10.13.624
              ABC STORES  STORE #522
                        ABC STORES  STORE #522
 100          gotad1 = cuad1;                                                   01       030207   7900 11.10.13.624
              231 70TH STREET
                       231 70TH STREET
 101          gotad2 = cuad2;                                                   01       030207   8000 11.10.13.624
 102          gotcity = cucity;                                                 01       030207   8100 11.10.13.624
              NEW YORK
                        NEW YORK
 103          except prtrec;                                                    01       030207   8200 11.10.13.624
 104         update custrec;  // update                                         01       031014   8300 11.10.13.624
CUCUST-0001000  CUSTOR-0000522  CUNAME-ABC  STORES    STORE  #522        CUAD1-231  70TH  STREET                      CUAD2-
CUCITY-NEW YORK                  CUSTA-NY CUZIP-10021
 105         endif;                                                            E01       030207   8400 11.10.13.624
 106       eval custor = 999;     // is no store 999                                     030207   8500 11.10.13.624
                   999
 107        chain custkey custmast;  // chain on file name                               030207   8600 11.10.13.624
           00010000000999
 108         if %found( custmast );                                            B01       030207   8700 11.10.13.624
 114         endif;                                                            E01       030207   9300 11.10.13.625
 115   // end of free formne                                                             060319   9400 11.10.13.625
 117 c                   eval      rorder = iorder +98  +                                020706   9600 11.10.13.625
                                 1690.043
                                            78.543
  97 c                             13.5 + PORDER                                         030207   9700 11.10.13.625
                                            1500
 119 c                   eval      cucust = 1000                                         030207   9800 11.10.13.625
                                     1000
 120 c     cucust        setll     custrec                                               030207   9900 11.10.13.625
           0001000
 121  * read by record name                                                              030207  10000 11.10.13.625
 122 c     cucust        reade     custrec                                84     eof     020710  10100 11.10.13.626
       N84 0001000
```

```
CUCUST-0001000  CUSTOR-0000000  CUNAME-ABC  STORES  INC.                     CUAD1-15  CORPORATE  DRIVE               CUAD2-
CUCITY-WEST CHESTER              CUSTA-PA CUZIP-19382
 123 c     *in84         doweq     *off                                        B01       020617  10200 11.10.13.626
           0
 124 c                   except    prtrec                                       01       030207  10300 11.10.13.626
 125 c     cucust        reade     custrec                                84    01       020617  10400 11.10.13.626
       N84 0001000
CUCUST-0001000  CUSTOR-0000001  CUNAME-ABC  STORES  INC                        CUAD1-423  MONTGOMERY  AVENUE          CUAD2-
CUCITY-ARDMORE                   CUSTA-PA CUZIP-19333
 123 c     *in84         doweq     *off                                        B01       020617  10200 11.10.13.626
           0
 124 c                   except    prtrec                                       01       030207  10300 11.10.13.626
 125 c     cucust        reade     custrec                                84    01       020617  10400 11.10.13.626
       N84 0001000
CUCUST-0001000  CUSTOR-0000002  CUNAME-ABC  STORES    STORE  #2           CUAD1-554  ARCH  STREET                     CUAD2-
CUCITY-PHILADELPHIA              CUSTA-PA CUZIP-19025
                                                                                                       11.10.13.595     PAGE
3
 123 c     *in84         doweq     *off                                        B01       020617  10200 11.10.13.626
           0
 124 c                   except    prtrec                                       01       030207  10300 11.10.13.626
 125 c     cucust        reade     custrec                                84    01       020617  10400 11.10.13.626
       N84 0001000
CUCUST-0001000  CUSTOR-0000522  CUNAME-ABC  STORES    STORE   #522        CUAD1-231  70TH  STREET                     CUAD2-
CUCITY-NEW YORK                  CUSTA-NY CUZIP-10021
 123 c     *in84         doweq     *off                                        B01       020617  10200 11.10.13.626
           0
 124 c                   except    prtrec                                       01       030207  10300 11.10.13.626
 125 c     cucust        reade     custrec                                84    01       020617  10400 11.10.13.626
        84 0001000
 126 c                   enddo                                                 E01       020617  10500 11.10.13.626
 127 c                   eval      *inlr=*on                                             030207  10600 11.10.13.626
                                   1
 128  * exit program                                                                     060319  10700 11.10.13.627
 129 c                   return                                                          030207  10800 11.10.13.627
```

RTPA for RPG Audit Output – NEWEXPSH (Interactive)

RPGLE – Fixed Format and Free Format RPG

Note – The RTPA Audit Output line is actually 198 positions, not 80 positions, so some of the audit output has been truncated.

Note – RTPA audit output records the exact moment in time that each statement is executed, and the contents of each field input by the online User and the User ID.

```
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.10.48.761 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-120906 TIMEN-111048
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.10.54.917 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111048
```

```
Program: NEWEXPSH  New Expected Ship Date RPGIV   Obj Lib: Z$AUDITE    Initiated:  12/09/06 11.10.48.733   PAGE   1
         NEWEXPSH    NEWEXPSH
Job: 026982               User Profile: PHH                          Source File/Library:  QRPGLESRC Z$AUDIT
Line#                                                                          Do# SrcId ChgDat   Seq#   Time
 956 C     *INZSR        BEGSR                                                     ph456 011227  89700 11.10.48.742
 957  * initialize fields and arrays                                                     060318  89800 11.10.48.742
 958 C                   MOVEL     *BLANKS       MOVSW             1               ph456 011227  89900 11.10.48.742
 959 C                   Z-ADD     12            $$D                               ph456 011227  90000 11.10.48.751
                                                 12121212121212121212121212
 960 C                   MOVEA     *ZERO         $$D2                              ph456 011227  90100 11.10.48.751
                                                 000000000000000000000000000000000000000
 961 C                   MOVEA     1111          $$D3                              ph457 011227  90200 11.10.48.751
                                                 1111000000000000000000000000000000000000000000000000
 962 C                   MOVEL     '88888888'    $$A                               ph456 011227  90300 11.10.48.751
 963 C                   ENDSR                                                     ph456 011227  90400 11.10.48.751
 229  * CUSTOMER MASTER KEY                                                              991225  17600 11.10.48.751
 234  * RECEIVE PARAMETERS PASSED FROM CALLING PROGRAM                                   990918  18100 11.10.48.751
 235 C     *ENTRY        PLIST                                                           990918  18200 11.10.48.751
 236 C                   PARM                    PARMIN            44                    991225  18300 11.10.48.751
                                                 000150000001
 238  * INPUT DATA AREA TSTDTA                                                           010126  18500 11.10.48.751
 240 C                   IN        TSTDTA                                                010126  18700 11.10.48.751
VAR TSTDTA                 1 -    100  ABCDEFGHIJKLMNOPQRSTUVWZYZ012345                    BBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
C
VAR TSTDTA              101 -    200  CCCCCCCCCCCCCCCCCCCCCCCCCCCCC                        DDDDDDDDDDDDDDDDDDDDDDDDDDDDDD
E
VAR TSTDTA           201 -   256  EEEEEEEEEEEEEEEEEEEEEEEEEEEEE                    FFFFFF
 242 C                   TIME                    TIMES             6 0                   000302  18900 11.10.48.751
                                                111048
 244  * MOVE INPUT PARM INTO FOUR PARM FIELDS IN DS                                      990918  19100 11.10.48.751
 245 C                   MOVEL     PARMIN        PARMRE                                  990918  19200 11.10.48.752
                                   000150000001
                                                 000150000001
 247  * start free form                                                                  061201  19301 11.10.48.752
 249   // now in free form RPG                                                           061201  19401 11.10.48.752
 250     torder = 1500;                                                                  020623  19500 11.10.48.752
           1500
 251       iorder = 78.543;                                                              020623  19600 11.10.48.752
           78.543
 252    // value of iorder has now been computed                                         061201  19601 11.10.48.752
 253         xorder = torder + 13.45  +                                                  020623  19700 11.10.48.753
            1618.19     1500
 254     // this is a continuation free form statement preceeded with +                  061201  19701 11.10.48.753
 255                   26.2 + iorder;                                                    020623  19800 11.10.48.753
                              78.543
 256   // end of free form                                                               061201  19801 11.10.48.753
 258  * resume fixed format calc statements                                              061201  19901 11.10.48.753
 259 c                   eval      rorder = iorder +98                                   020623  20000 11.10.48.753
                                  176.543
                                            78.543
 260  * add 30 days to start_date to get end_date                                        020623  20100 11.10.48.753
 261 c     start_date    adddur    30:*days      end_date                                020623  20200 11.10.48.753
           1998-12-18                            1999-01-17
 262  * add 1 month to end_date                                                          020623  20300 11.10.48.753
 263 c                   adddur    1:*months     end_date                                020623  20400 11.10.48.753
                                                 1999-02-17
 264  * extract day number from date                                                     050516  20500 11.10.48.754
 265 c                   extrct    end_date:*D   dayno             2 0          dd       050516  20600 11.10.48.754
                                   1999-02-17
                                                    17
 266  * extract month number from date                                                   050517  20700 11.10.48.754
 267 c                   extrct    end_date:*M   Month_no          2 0          mm       050517  20800 11.10.48.754
                                   1999-02-17
                                                         2
 268  * extract year number from date                                                    050517  20900 11.10.48.754
 269 c                   extrct    end_date:*Y   Year_no           4 0          ccyy     050517  21000 11.10.48.754
                                   1999-02-17
                                                    1999
 270  * add 1 year to start_date                                                         020623  21100 11.10.48.754
 271 c     employ_dat    adddur    1:*years      anniv_dat                               020623  21200 11.10.48.754
           12/29/1992                            12/29/1993
 272  * add 3 hours, 22 minutes and 50 seconds to midnight                               020623  21300 11.10.48.754
 273 c     T'00.00.00'   adddur    3:*hours      end_time                                020623  21400 11.10.48.754
                                                 03:00:00
 274 c                   adddur    22:*minutes   end_time                                020623  21500 11.10.48.754
                                                 03:22:00
 275 c                   adddur    50:*seconds   end_time                                020623  21600 11.10.48.754
                                                 03:22:50
 276  * add 1000 microseconds to a time stamp (26 character date and time)               020623  21700 11.10.48.754
 277 c                   adddur    1000:*ms      total_time                              020623  21800 11.10.48.758
                                                 0001-01-01-00.00.00.001000
```

```
                                                                                                       11.10.48.733   PAGE
 278 C                   Z-ADD     14.25         TESD             12 3                   010113  21900 11.10.48.758
                                               14.250
 279 C                   Z-ADD     *zero         LocatTotal                              050415  22000 11.10.48.758

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 281  * MOVE INPUT ORDER AND LINE TO KORDER AND LINE (KEYED)                             011227  22200 11.10.48.760
 282 C                   Z-ADD     PORDER        KORDER                                  020623  22300 11.10.48.760
                                     1500
                                                   1500
 283 C                   Z-ADD     PLINE         KLINE                                   001007  22400 11.10.48.761
                                       1
                                                     1
 284  *---------------------------------------------------------------                  000323  22500 11.10.48.761
 285  * DISPLAY HEADING SCREEN                                                           000323  22600 11.10.48.761
 286  *---------------------------------------------------------------                  000323  22700 11.10.48.761
 287 C     DISP01        TAG                                                             000514  22800 11.10.48.761
 288  * CLEAR EXPECTED SHIP DATE AND ERROR CODE                                          000514  22900 11.10.48.761
 289 C                   Z-ADD     *ZERO         PEXPSH                                  000514  23000 11.10.48.761
                                                      0
 290 C                   MOVEL     *BLANKS       PERROR                                  000514  23100 11.10.48.761
 291 C                   Z-ADD     *ZEROS        KCUSNO                                  001002  23200 11.10.48.761
                                                      0
 292 C                   Z-ADD     *ZEROS        KSTORE                                  001002  23300 11.10.48.761
                                                      0
 293 C                   MOVEL     *BLANKS       KCUSNA                                  000323  23400 11.10.48.761
 294 C                   Z-ADD     *ZERO         EXPMDY                                  000323  23500 11.10.48.761
                                                      0
 295 C                   TIME                    TIMEN             6 0                   010501  23600 11.10.48.761
                                                111048
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.10.48.761 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00001 UDATE-120906 TIMEN-111048
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.10.54.917 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111048
 297  * TEST F3                                                                          000323  23800 11.10.54.917
 298 C     *IN03         CABEQ     *ON           DONE                                    000323  23900 11.10.54.917
           0
 300 C     UDATE         CABEQ     090100        DONE                                    010113  24100 11.10.54.917
          120906
 302  *---------------------------------------------------------------                  000323  24300 11.10.54.917
 303  * VALIDATE ORDER # AND LINE #                                                      000323  24400 11.10.54.917
 304  *---------------------------------------------------------------                  000323  24500 11.10.54.917
 306  * GET ORDER DETAIL RECORD FOR ORDER# AND LINE#                                     990918  24700 11.10.54.917
 307 C                   Z-ADD     KORDER        OORDER                                  001007  24800 11.10.54.917
                                     1500
                                                   1500
 308 C                   Z-ADD     KLINE         OLINE                                   001007  24900 11.10.54.918
                                       2
                                                     2
 309  *--------------------------------------------------------------------             000909  25000 11.10.54.918
 310  **** ORDKEY        CHAIN     ODETREC                            25        25       000909  25100 11.10.54.918
 311  **** *IN25         CABEQ     *ON           DISP01                   42             000909  25200 11.10.54.918
 313  * NO INDICATOR USED ON CHAIN  chain by file name, not record name                  030426  25400 11.10.54.918
 314  * AUDIT RPGIV CHAIN STATEMENT (NO ERROR INDICATOR)                                 000918  25500 11.10.54.918
 315 C     ordkey        chain     orderde                                  25 ph234 030504  25600 11.10.54.931
           000150000002
ODORD#-0001500  ODLINE-00002  ODCUST-0001000  ODSTOR-0000001  ODITEM-Y2430        ODPRIC-0002515  ODQTY-0000003  ODREQD-20000317
ODEXPD-20051126 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
 316 C                   if        not%found                             B01 ph234 030504  25700 11.10.54.931
 319 C                   END                                             E01 ph235 030504  26000 11.10.54.931
 320  *--------------------------------------------------------------------             000909  26100 11.10.54.931
 322  * DID GET ORDER DETAIL RECORD                                                      000323  26300 11.10.54.931
 323  * CONVERT ODEXPD FORMAT YYYYMMDD TO PEXPSH FORMAT MMDDYY                           000317  26400 11.10.54.931
 324 C     odexpd        ifne      *zero                                 B01 ph543 030504  26500 11.10.54.931
         20051126
 325 C                   z-add     odexpd        expmd             4 0          01 ph543 030504  26600 11.10.54.931
                                 20051126
                                                  1126
 326 C     odexpd        DIV       10000         expyy             2 0                   001029  31100 11.10.54.931
         20051126                                    5
 327 C     expmd         mult      100           expmdy            6 0          01       010330  26800 11.10.54.932
            1126                                 112600
 328 C     expmdy        add       expyy         expmdy                         01       010330  26900 11.10.54.932
           112605                      5
                                                 112605
 329 C                   endif                                                 E01       010330  27000 11.10.54.932
 331  *----------------------------------------------------------------------            000917  27200 11.10.54.932
 332  * AUDIT RPGIV EVAL STATEMENT (LOWER CASE)                                          010113  27300 11.10.54.932
 333 C                   z-add     *zero         answer            7 0                   010113  27400 11.10.54.932
                                                      0
                                                                                                       11.10.48.733   PAGE
 334 C                   eval                    answer = expmdy + expyy                 010522  27500 11.10.54.932
                                                 112610   112605       5
 335  *----------------------------------------------------------------------            010522  27600 11.10.54.932
 336  * AUDIT RPGIV eval starting in extended factor 2                                   010522  27700 11.10.54.932
 337 C                   eval      answer = expmdy + expyy                               010522  27800 11.10.54.932
                                   112610   112605       5
 338  *----------------------------------------------------------------------            000917  27900 11.10.54.932
 339 C                   EVAL      @STSC = *BLANKS                                       010730  28000 11.10.54.932
 342 C                   movel     '1'           @yes              1                     010730  28300 11.10.54.932
                                                 1
 343 C                   movel     '1'           @1stline          1                     010730  28400 11.10.54.934
                                                 1
 344 C                   setoff                                       33                 010730  28500 11.10.54.934
 345 C                   IF        @1stline = @yes and                         B01       010730  28600 11.10.54.934
                                   1          1
 346 C                             *IN33 = *OFF AND                            B01       050118  28700 11.10.54.934
                                   0
 347 C                             *INLR = *OFF                                B01       050118  28800 11.10.54.934
                                   0
 348 C                   movel     '2'           hold2             1            01       010730  28900 11.10.54.938
                                                 2
 349 C                   ENDIF                                                 E01       010730  29000 11.10.54.938
 351 C                   dou       @1stline = @yes or                          B01 PDDD  010814  29200 11.10.54.938
 308 C                             *IN33 = *OFF                                B01 PDDD  010814  29300 11.10.54.938
 353 C                   movel     '3'           hold2             1            01       010730  29400 11.10.54.938
                                                 3
 354 C                   enddo                                                 E01 PDDD  010814  29500 11.10.54.938
 355 C                   movel     '4'           hold2             1                     010730  29600 11.10.54.938
                                                 4
 356  * AUDIT RPGIV                                                                      010522  29700 11.10.54.938
 357 C                   eval      *IN50 = *ON                                           010522  29800 11.10.54.938
                                   1
 358 C                   eval                 *in55 = *OFF                               010710  29900 11.10.54.938
                                              0
 359 C                   eval                   *IN55 = *OFF                             010710  30000 11.10.54.938
                                                0
 360 C                   eval                     *in(60) = *ON                          010710  30100 11.10.54.938
                                                  1
 361 C                   eval                       *IN(60) = *ON                        010710  30200 11.10.54.938
                                                    1
 362 C                   eval                *IN = *OFF                                  010710  30300 11.10.54.939

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 363  *----------------------------------------------------------------------            010522  30400 11.10.54.939
 364  * AUDIT RPGIV DOW STATEMENT                                                        000918  30500 11.10.54.939
 365 C                   Z-ADD     *ZERO         COUNTER           2 0                   000918  30600 11.10.54.939
                                                       0
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.10.54.939
                                                       0
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.10.54.939
                                                       1
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.10.54.939
                                                       1
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.10.54.939
                                                       2
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.10.54.939
                                                       2
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.10.54.939
                                                       3
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.10.54.939
                                                       3
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.10.54.939
                                                       4
```

```
366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.10.54.939
                                                      4
367 C                   ADD       1             COUNTER                        01       000918  30800 11.10.54.939
                                                      5
366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.10.54.939
                                                      5
367 C                   ADD       1             COUNTER                        01       000918  30800 11.10.54.939
                                                      6
368 C                   ENDDO                                                 E01       000918  30900 11.10.54.939
369  *---------------------------------------------------------------------             000918  31000 11.10.54.939
370  * AUDIT RPGIV select, WHEN, OTHER STATEMENTS                                       001029  31100 11.10.54.939
371 C                   SELECT                                                B01       001029  31200 11.10.54.939
372 C                   WHEN      COUNTER = 6                                 X01       001029  31300 11.10.54.939
                                        6
372 C                   WHEN      COUNTER = 6                                 X01       001029  31300 11.10.54.939
                                        6
                                                                                                      11.10.48.733   PAGE
373 C                   Z-ADD     *ZERO         COUNTER                        01       001029  31400 11.10.54.939
                                                      0
375 C                   ENDSL                                                 E01       001029  31600 11.10.54.939
376  * AUDIT RPGIV select, WHEN, OTHER STATEMENTS                                       010912  31700 11.10.54.940
377 C                   movel     'P'           @mode             1                     010912  31800 11.10.54.940
                                                P
378 C                   z-add     1             PHSCNO            1 0                   010912  31900 11.10.54.940
                                                     1
379 C                   SELECT                                                B01       010912  32000 11.10.54.940
380 C                   WHEN      @mode = 'P'                                 X01       010912  32100 11.10.54.940
                                  P
381 C                              OR PHSCNO = 1                              X01       010912  32200 11.10.54.940
                                           1
380 C                   WHEN      @mode = 'P'                                 X01       010912  32100 11.10.54.940
                                  P
337 C                              OR PHSCNO = 1                              X01       010912  32200 11.10.54.940
                                           1
382 C                   Z-ADD     *ZERO         COUNTER                        01       010912  32300 11.10.54.941
                                                      0
385 C                   ENDSL                                                 E01       010912  32600 11.10.54.941
386  *---------------------------------------------------------------------             010912  32700 11.10.54.941
387 C                   if        counter <> 0                                B01       010912  32800 11.10.54.941
                                        0
389 C                   endif                                                 E01       010912  33000 11.10.54.941
390 C                   movel     'R'           @mode             1                     010912  33100 11.10.54.941
                                                R
391 C                   movel     '3'           RTYP              1                     010912  33200 11.10.54.942
                                                3
392 C                   SELECT                                                B01       010912  33300 11.10.54.942
393 C                   WHEN      @mode = 'R'                                 X01       010912  33400 11.10.54.942
                                  R
394 C                             AND RTYP = '3'                              X01       010912  33500 11.10.54.942
                                      3
393 C                   WHEN      @mode = 'R'                                 X01       010912  33400 11.10.54.943
                                  R
350 C                             AND RTYP = '3'                              X01       010912  33500 11.10.54.943
                                      3
395 C                   Z-ADD     *ZERO         COUNTER                        01       010912  33600 11.10.54.943
                                                      0
398 C                   ENDSL                                                 E01       010912  33900 11.10.54.943
399  *---------------------------------------------------------------------             001029  34000 11.10.54.943
400  * AUDIT RPGIV IF STATEMENT                                                         001029  34100 11.10.54.943
401 C                   IF        COUNTER = 0                                 B01       001029  34200 11.10.54.943
                                        0
402 C                   Z-ADD     3             COUNTER                        01       001029  34300 11.10.54.943
                                                      3
403 C                   ENDIF                                                 E01       001029  34400 11.10.54.943
404  *---------------------------------------------------------------------             001029  34500 11.10.54.943
405 C                   if        counter > 0                                 B01       010522  34600 11.10.54.943
                                        3
406 C                   endif                                                 E01       010522  34700 11.10.54.943
407  *---------------------------------------------------------------------             010522  34800 11.10.54.943
408 C                   if        counter <> 0                                B01       010522  34900 11.10.54.943
                                        3
409 C                   endif                                                 E01       010522  35000 11.10.54.944
```

```
410  *---------------------------------------------------------------------             010522  35100 11.10.54.944
411  * AUDIT RPGIV EXTENDED FACTOR 2 CONDITIONAL AND/OR COMPLEX STATEMENTS             010602  35200 11.10.54.944
412 C                   Z-ADD     2             COUNTER                                  010602  35300 11.10.54.944
                                                      2
413 C                   Z-ADD     7             answer                                   010602  35400 11.10.54.944
                                                     7
414 C                   Z-ADD     *zero         final             7 2                   010607  35500 11.10.54.944
                                                  .00
415 C                   Z-sub     *zero         sum               6 1                   010704  35600 11.10.54.944
                                                 .0
416 C                   Z-ADD     3115          total             8 0                   010604  35700 11.10.54.944
                                                 3115
417 C                   Z-ADD     112           net               3 0                   010607  35800 11.10.54.944
                                                112
418 C                   IF        COUNTER = 0                                 B01       010602  35900 11.10.54.944
                                        2
419 C                             OR COUNTER = 2                              B01       010602  36000 11.10.54.944
                                           2
420 C                             OR COUNTER = 4                              B01       010602  36100 11.10.54.944
                                           2
421 C                             AND ANSWER = 7                              B01       010602  36200 11.10.54.944
                                           7
446 C                                     CCCC - DIFFERENCE +160 + EXTRA +              010617  38700 11.10.54.944
                                           .00          .00            0
447 C                                     188  - INTERIM + EXTRA2                       010617  38800 11.10.54.944
                                                     .00        0
448 C                                   - 33.15 + GROSS                                 010617  38900 11.10.54.944
                                                    .00
449 C                                   + MORE                                          010617  39000 11.10.54.944
                                           .00
452 C                             + 44 - GROSS - EXTRA2                                 010617  39300 11.10.54.944
                                           .00        0
                                                                                                      11.10.48.733   PAGE
422 C                   Z-ADD     3             COUNTER                        01       010604  36300 11.10.54.963
                                                      3
423 C                   ENDIF                                                 E01       010602  36400 11.10.54.963
424 C     sum           add       total         final                                   010604  36500 11.10.54.963
            .0                     3115
                                              3115.00
425 C                   eval      final = answer + counter + 5                          010602  36600 11.10.54.964
                                  15.00        7         3
426 C                   eval      sum = 4 + 6 - 2 + 1555                                010602  36700 11.10.54.964
                               1563.0
427 C                   eval      final = answer * counter                              010610  36800 11.10.54.964
                                  21.00        7         3
428 C                   eval      final = answer* counter                               010610  36900 11.10.54.964
                                  21.00        7        3
429 C                   eval      final = answer *counter                               010610  37000 11.10.54.964
                                  21.00        7        3
430 C                   eval      final = answer*counter                                010610  37100 11.10.54.964
                                  21.00        7       3
431 C                   eval      final = answer / counter                              010610  37200 11.10.54.964
                                   2.33        7         3
432 C                   eval      final = answer/counter                                010610  37300 11.10.54.964
                                   2.33        7       3
433 C                   eval      sum = 4 + 6 - counter + 1555                          010605  37400 11.10.54.964
                               1562.0                 3
434 C                   eval        total = 4 + 6 + sum                                 010603  37500 11.10.54.965
                                     1572        1562.0
435 C                   eval                    sum = 4 + 6                             010603  37600 11.10.54.965
                                               10.0
436 C                   eval      total = 4 + sum + 6 + answer + final - net            010606  37700 11.10.54.965
                                     82-     10.0            7
                                                                  2.33   112
437  *---------------------------------------------------------------------             010602  37800 11.10.54.965
438 C                   Z-ADD     7.12          cccc              5 2                   010617  37900 11.10.54.965
                                                7.12
439 C                   Z-ADD     163.23        gross             6 2                   010617  38000 11.10.54.965
                                               163.23
440 C                   z-add     1444.2        difference        8 2                   010617  38100 11.10.54.967
                                                   1444.20
441 C                   z-add     43.8          interim           8 2                   010617  38200 11.10.54.967
```

```
                                                   43.80
 442 C                   z-add     87            extra            5 0                   010617  38300 11.10.54.967
                                                    87
 443 C                   z-add     105           extra2           5 0                   010617  38400 11.10.54.967
                                                    105
 444 C                   z-add     17            more             8 2                   010617  38500 11.10.54.967
                                                17.00
 445 C                   eval      final = answer + counter +                           010617  38600 11.10.54.967
                                  783.80-      7         3
 402 C                                     CCCC - DIFFERENCE +160 + EXTRA +             010617  39000 11.10.54.967
                                           7.12      1444.20           87
 403 C                                     188  - INTERIM + EXTRA2                      010617  39000 11.10.54.967
                                                    43.80      105
 404 C                                   - 33.15 + GROSS                                010617  39000 11.10.54.967
                                                  163.23
 405 C                                   + MORE                                         010617  39000 11.10.54.967
                                          17.00
 450 C                   eval      net  = counter + cccc                                010617  39100 11.10.54.969
                                    10          3   7.12
 451 C                   eval      interim = answer - cccc + net + difference           010617  39200 11.10.54.969
                                   1229.85        7   7.12    10      1444.20
 408 C                             + 44 - GROSS - EXTRA2                                010617  39300 11.10.54.969
                                         163.23      105
 453  *-----------------------------------------------------------------------          010617  39400 11.10.54.969
 454 C                   movel     '2'           movsw            1                     010604  39500 11.10.54.969
                                                 2
 455  * AUDIT RPGIV IF STATEMENT with alpha extended factor 2                           010604  39600 11.10.54.971
 456 C                   if        movsw = '2'                               B01        010604  39700 11.10.54.971
                                   2
 457 C                             OR MOVSW = '3'                            B01        010604  39800 11.10.54.971
                                      2
 458 C                   movel     '1'           movsw            1           01        010604  39900 11.10.54.971
                                                 1
 459 C                   endif                                               E01        010604  40000 11.10.54.971
 460 C                   movel     '5'           movsw            1                     010604  40100 11.10.54.971
                                                 5
 461 C                   if        movsw = '5'                               B01        010604  40200 11.10.54.972
                                   5
 462 C                   endif                                               E01        010604  40300 11.10.54.972
 463  *-----------------------------------------------------------------------          010604  40400 11.10.54.972
                                                                                                      11.10.48.733     PAGE
6
 464 C                   Z-ADD     2             COUNTER                                010615  40500 11.10.54.972
                                                       2
 465 C                   Z-ADD     7             answer                                 010615  40600 11.10.54.972
                                                      7
 466 C                   Z-ADD     14.2          gggggg           6 2                   010615  40700 11.10.54.972
                                                  14.20
 467 C                   IF        COUNTER = 0 and  gggggg =5 and                    B01        010615  40800 11.10.54.972
                                         2         14.20
 468 C                                COUNTER > 5 AND                            B01        010615  40900 11.10.54.972
                                            2
 469 C                                COUNTER < 3 OR ANSWER = 6 AND              B01        010615  41000 11.10.54.972
                                            2             7
 470 C                                COUNTER < 2                                B01        010615  41100 11.10.54.972
                                            2
 471 C                             OR COUNTER = 2                                B01        010615  41200 11.10.54.972
                                            2
 472 C                             AND ANSWER = 7 OR                             B01        010615  41300 11.10.54.972
                                            7
 473 C                             ANSWER = 5                                    B01        010615  41400 11.10.54.972
                                        7
 474 C                   Z-ADD     3             COUNTER                          01        010615  41500 11.10.54.987
                                                       3
 475 C                   ENDIF                                               E01        010615  41600 11.10.54.987
 476  *                                                                -----            010615  41700 11.10.54.987
 477  *-----------------------------------------------------------------------          010615  41800 11.10.54.987
 478  * AUDIT RPGIV MOVEL PADDED WITH BLANKS STATEMENT                                  010522  41900 11.10.54.987
 479 C                   MOVEL     *ALL'M'       TESTML          20                     001029  42000 11.10.54.987
                                                 MMMMMMMMMMMMMMMMMMMM
 480 C                   MOVEL(P)  'LEFT'        TESTML                                 001029  42100 11.10.54.987
                                                 LEFT
 481  *---------------------------------------------------------------------             010330  42200 11.10.54.987
 482  * all lower case statements                                                        010330  42300 11.10.54.987
 483 C                   movel     *ALL'L'       testlo           20                     010330  42400 11.10.54.987
                                                 LLLLLLLLLLLLLLLLLLLL
 484 C                   z-add     11111         aaaaaaaaaa        6 0                   010330  42500 11.10.54.987
                                                      11111
 485 C                   z-add     2222222       bbbbbbbbbb        8 3                   010618  42600 11.10.54.988
                                                  22222.000
 486 C     aaaaaaaaaa    add       bbbbbbbbbb    cccccccccc        8 0                   010330  42700 11.10.54.988
                11111               22222.000
                                                      33333
 487  *---------------------------------------------------------------------             001029  42800 11.10.54.988
 488  * AUDIT RPGIV TIME STATEMENT                                                       001029  42900 11.10.54.988
 489 C                   TIME                    TIMENOW           6 0                   001029  43000 11.10.54.988
                                                  111054
 490  *---------------------------------------------------------------------             001029  43100 11.10.54.988
 491 C                   movel     '2'           movsw             1                     010530  43200 11.10.54.988
                                                 2
 492  * AUDIT RPGIV IF STATEMENT                                                         010113  43300 11.10.54.988
 493 C                   if        movsw = *blanks                             B01       010113  43400 11.10.54.988
                                   2
 494 C                             OR COUNTER = 0                              B01       010113  43500 11.10.54.988
                                            3
 496 C                   endif                                                 E01       010113  43700 11.10.54.989
 497  *---------------------------------------------------------------------             010113  43800 11.10.54.989
 499 C     100           DIV       5.25          NET               3 0          01       000323  48400 11.10.54.989
                                                  19
 500 C                   MVR                     FRACT             4 4          01       000323  48500 11.10.54.989
                                                 .2500
 502  *  GOT ORDER DETAIL, PRINT IT                                                      000514  44300 11.10.54.989
 503 C                   EXCEPT    PRTDET                              234               000514  44400 11.10.54.989
 505  * GET THE CUSTOMER MASTER STORE RECORD                                             991225  44600 11.10.54.989
 506 C                   Z-ADD     ODCUST        CUCUST                                  991225  44700 11.10.54.989
                                     1000
                                                   1000
 507 C                   Z-ADD     ODSTOR        CUSTOR                                  991225  44800 11.10.54.990
                                        1
                                                      1
 508  *----------------------------------------------------------------                  010118  44900 11.10.54.990
 509  * THIS ROUTINE CAUSES THE WRONG CUSTOMER TO BE DISPLAYED                           010118  45000 11.10.54.990
 510  * IF THE ORDER DETAIL FILE HAS BEEN UPDATED TWICE, AND ONLY TWICE                  010118  45100 11.10.54.990
 511  * FIND THE ERROR ON THE AUDIT REPORT BY SCANNING FOR 2050                          010118  45200 11.10.54.990
 512 C     UPDREC        IFEQ      2                                           B01       010118  45300 11.10.54.990
                0
 521 C                   ENDIF                                                 E01       010118  46200 11.10.54.990
 523  *----------------------------------------------------------------                  010118  46400 11.10.54.990
 524 C     CUSKEY        CHAIN     CUSTREC1                           30                 000717  46500 11.10.54.990
       N30 00010000000001
CUCUST-0001000  CUSTOR-0000001  CUNAME-ABC  STORES  INC                        CUAD1-423  MONTGOMERY  AVENUE          CUAD2-
CUSTA-PA
                                                                                                       11.10.48.733   PAGE
 525 C                   z-add     *all'1'       aa                3 0                   050102  46600 11.10.54.991
                                                111
 526 C                   z-add     *all'2'       bb                3 0                   050102  46700 11.10.54.991
                                                222
 527 C                   z-add     *all'3'       cc                3 0                   050102  46800 11.10.54.991
                                                333
 528 C                   z-add     *all'4'       dd                3 0                   050102  46900 11.10.54.991
                                                444
 529 C                   z-add     *all'5'       ee                3 0                   050102  47000 11.10.54.991
                                                555
 530 C                   z-add     *all'6'       ff                3 0                   050102  47100 11.10.54.991
                                                666
 531 C                   z-add     *all'7'       gg                3 0                   050102  47200 11.10.54.991
                                                777
 532 C                   z-add     *all'8'       hh                3 0                   050102  47300 11.10.54.991
                                                888
 533 C                   z-add     *all'9'       ii                3 0                   050102  47400 11.10.54.991
                                                999
 534 C                   z-add     *zeros        total             8 0                   050102  47500 11.10.54.991
                                                     0
 536   total = aa + bb + cc + dd + ee + ff + gg + hh + ii;                               050102  47700 11.10.54.991
```

```
        4995  111  222  333  444  555  666  777  888  999
 538 c                   eval      total = aa + bb + cc + dd + ee + ff + gg + hh         050102  47900 11.10.54.991
                                    3996  111  222  333  444  555  666  777  888
 539 C     *IN30         IFEQ      *OFF                                        B01       991225  48000 11.10.54.992
           0
 540  * GOT CUSTOMER MASTER                                                              991225  48100 11.10.54.992
 541 C                   Z-ADD     CUCUST        KCUSNO                         01       001007  48200 11.10.54.992
                                     1000
                                                   1000
 542 C                   Z-ADD     CUSTOR        KSTORE                         01       010118  48300 11.10.54.992
                                        1
                                                      1
 543 C                   MOVEL     CUNAME        KCUSNA                         01       000323  48400 11.10.54.992
                                   ABC STORES INC
                                                 ABC STORES INC
 544 C                   MOVEL     CUNAME        PCUSNA                         01       000323  48500 11.10.54.992
                                   ABC STORES INC
                                                 ABC STORES INC
 545 C                   EXCEPT    PRTCUS                                       01       000323  48600 11.10.54.992
 548 C                   ENDIF                                         E01       991225  48900 11.10.54.992
 550  * DISPLAY DETAIL SCREEN                                                            000323  49100 11.10.54.993
 551 C     DISP02        TAG                                                             000514  49200 11.10.54.993
 552 C                   TIME                    TIMEN             6 0                   010501  49300 11.10.54.993
                                                111054
 553 C                   EXFMT     NEWEXPD2                                              051007  49400 11.10.54.993 WRITE
*IN03-0 *IN43-0 EXPMDY-112605 KCUSNO-0001000 KCUSNA-ABC STORES INC                UDATE-120906 KSTORE-0000001 TIMEN-111054
KORDER-0001500 KLINE-00002
 553 C                   EXFMT     NEWEXPD2                                              051007  49400 11.11.11.149 READ
*IN03-0 *IN43-0 EXPMDY-011807 KCUSNO-0001000 KCUSNA-ABC STORES INC                UDATE-120906 KSTORE-0000001 TIMEN-111054
KORDER-0001500 KLINE-00002
 554  * TEST F3                                                                          000323  49500 11.11.11.149
 555 C     *IN03         CABEQ     *ON           DONE                                    000323  49600 11.11.11.149
           0
 557  * VALIDATE CHANGED DATE, AND UPDATE ORDER DETAIL                                   000323  49800 11.11.11.150
 559  * CONVERT EXPMDY FORMAT MMDDYY TO ODEXPD FORMAT YYYYMMDDYY                         000323  50000 11.11.11.150
 561 C                   Z-ADD     EXPMDY        YY                2 0          YY       000323  50200 11.11.11.150
                                    11807
                                                  7
 562 C     EXPMDY        DIV       100           MMDD              4 0          MMDD     000323  54700 11.11.11.150
            11807                                 118
 563 C     YY            MULT      10000         Y4MMDD            8 0          00YY0000 000323  50400 11.11.11.150
            7                                     70000
 564 C                   ADD       MMDD          Y4MMDD                         00YYMMDD 000323  50500 11.11.11.150
                                    118
                                                  70118
 565 C     YY            IFGT      40                                          B01       000323  50600 11.11.11.150
            7
 567 C                   ELSE                                                  X01       000323  50800 11.11.11.152
 568 C                   ADD       20000000      Y4MMDD                         01       000323  50900 11.11.11.152
                                               20070118
 569 C                   END                                                   E01       000323  51000 11.11.11.152
 570 C                   Z-ADD     Y4MMDD        ODEXPD                                  000323  51100 11.11.11.152
                                 20070118
                                               20070118
 571  * COMPLEX IF STATEMENT                                                             010614  51200 11.11.11.152
 572 C                   IF        COUNTER = 0 and                             B01       010614  51300 11.11.11.152
                                         3
 573 C                                COUNTER > 5 AND                          B01       010614  51400 11.11.11.152
                                            3
 574 C                                COUNTER < 3 OR ANSWER = 6 AND            B01       010614  51500 11.11.11.152
                                            3             7
 575 C                                COUNTER < 2                              B01       010614  51600 11.11.11.152
                                            3
 576 C                             OR COUNTER = 2                              B01       010614  51700 11.11.11.152
                                            3
 577 C                             AND ANSWER = 7 OR                           B01       010614  51800 11.11.11.152
                                            7
 578 C                             ANSWER = 5                                  B01       010614  51900 11.11.11.152
                                        7
                                                                                                       11.10.48.733   PAGE
 580 C                   ENDIF                                                 E01       010614  52100 11.11.11.153
 582  * TEST FOR FIELD OVER 100...                                                       010411  52300 11.11.11.153
 583 C                   MOVE      *ALL'#'       ALL#            256                     010411  52400 11.11.11.153
VAR     ALL#                                                                                      1     -                100
####################################################################################################
VAR      ALL#                                                                                 101      -                 200
####################################################################################################
VAR ALL#              201 -   256  ########################################################
 584 C                   MOVE      ALL#          ALL$            256                     010411  52500 11.11.11.153
VAR     ALL#                                                                                      1     -                100
####################################################################################################
VAR      ALL#                                                                                 101      -                 200
####################################################################################################
VAR ALL#              201 -   256  ########################################################
VAR     ALL$                                                                                      1     -                100
####################################################################################################
VAR      ALL$                                                                                 101      -                 200
####################################################################################################
VAR ALL$              201 -   256  ########################################################
 586 C                   MOVE      '2'           MOVSW                                   010429  52700 11.11.11.153
                                                 2
 587 C                   MOVE      '3'           MOVSW1            1                     010429  52800 11.11.11.153
                                                 3
 588 C                   MOVE      '4'           MOVSW2            1                     010429  52900 11.11.11.153
                                                 4
 589 C                   MOVE      '5'           MOVSW3            1                     010429  53000 11.11.11.153
                                                 5
 590 C                   MOVE      '5'           MOVSW4            1                     010429  53100 11.11.11.154
                                                 5
 591 C                   MOVE      'A'           HLD1              1                     010429  53200 11.11.11.154
                                                 A
 593 C     EXPMDY        IFEQ      UDATE                                       B01       010420  53400 11.11.11.154
            11807                  120906
 594 C     MOVSW         ANDEQ     *BLANK                                       01       000514  53500 11.11.11.154
           2
 595 C     MOVSW         OREQ      '5'                                          01       010429  53600 11.11.11.154
           2
 596 C     MOVSW1        OREQ      MOVSW2                                       01       010429  53700 11.11.11.154
           3                       4
 597 C     MOVSW3        ORNE      MOVSW4                                       01       010429  53800 11.11.11.154
           5                       5
 598 C     MOVSW         OREQ      '8'                                          01       010429  53900 11.11.11.154
           2
 599 C     MOVSW         OREQ      '9'                                          01       010429  54000 11.11.11.154
           2
 600 C     MOVSW         OREQ      'C'                                          01       010429  54100 11.11.11.154
           2
 601 C     MOVSW         OREQ      'D'                                          01       010429  54200 11.11.11.154
           2
 602 C     MOVSW         OREQ      HLD1                                         01       010429  54300 11.11.11.154
           2                       A
 603 C     MOVSW3        ANDNE     MOVSW2                                       01       010429  54400 11.11.11.154
           5                       4
 605 C                   END                                               E01       000323  54600 11.11.11.156
 607  * UPDATE ORDER DETAIL EXPECTED SHIP DATE                                           000323  54800 11.11.11.156
 608 C                   ADD       1             UPDREC            6 0                   010118  54900 11.11.11.156
                                                      1
 609 C                   UPDATE    ODETREC                                               000323  55000 11.11.11.156
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430        ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317
ODEXPD-20070118 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
 611  * WRITE UPDATED ORDER DETAIL TO A WORK FILE                                        000402  55200 11.11.11.160
 612 C                   CLEAR                   ODETWRK                                 000402  55300 11.11.11.160
 613 C                   Z-ADD     ODORD#        WDORD#                                  000402  55400 11.11.11.160
                                     1500
                                                   1500
 614 C                   Z-ADD     ODLINE        WDLINE                                  000402  55500 11.11.11.160
                                        2
                                                      2
 615 C                   Z-ADD     ODCUST        WDCUST                                  000402  55600 11.11.11.160
                                     1000
                                                   1000
 616 C                   Z-ADD     ODSTOR        WDSTOR                                  000402  55700 11.11.11.160
                                        1
                                                      1
 617 C                   MOVEL     ODITEM        WDITEM                                  000402  55800 11.11.11.160
                                   Y2430
                                                 Y2430
 618 C                   Z-ADD     ODPRIC        WDPRIC                                  000402  55900 11.11.11.160
                                    25.15
                                                  25.15
 619 C                   Z-ADD     ODQTY         WDQTY                                   000402  56000 11.11.11.161
                                       3
                                                     3
 620 C                   Z-ADD     ODREQD        WDREQD                                  000402  56100 11.11.11.162
                                 20000317
                                               20000317
                                                                                                       11.10.48.733   PAGE
 621 C                   Z-ADD     ODEXPD        WDEXPD                                  000402  56200 11.11.11.162
                                 20070118
                                               20070118
 622 C                   Z-ADD     ODSHPD        WDSHPD                                  000402  56300 11.11.11.163
                                        0
                                                      0
 623 C                   Z-ADD     ODINV#        WDINV#                                  000402  56400 11.11.11.163
                                        0
                                                      0
 624 C                   MOVEL     ODSTAT        WDSTAT                                  000402  56500 11.11.11.163
                                   O
                                                 O
 625 C                   MOVEL     ODX           WDX                                     000402  56600 11.11.11.163
 627  * WRITE ORDERWK                                                                    000402  56800 11.11.11.163
 628 C                   WRITE     ODETWRK                                               000402  56900 11.11.11.163
WDORD#-0001500 WDLINE-00002 WDCUST-0001000 WDSTOR-0000001 WDITEM-Y2430                 WDPRIC-0002515 WDQTY-0000003 WDREQD-
20000317 WDEXPD-20070118 WDSHPD-00000000 WDINV#-0000000 WDSTAT-O WDX-
 630  * NESTED IF STATEMENTS WITH AND/OR                                                 010429  57100 11.11.11.163
 631 C     aaaaaaaaaa    ifeq      bbbbbbbbbb                                   B01      010429  57200 11.11.11.164
                11111               22222.000
 632 C     UPDREC        andeq     YY                                            01      010429  57300 11.11.11.164
                1                   7
 637 C                   END                                                    E01      010429  57800 11.11.11.164
 639  * REDISPLAY FIRST SCREEN                                                           000323  58000 11.11.11.164
 640 C                   GOTO      DISP01                                                000514  58100 11.11.11.164
 287 C     DISP01        TAG                                                             000514  22800 11.11.11.164
 288  * CLEAR EXPECTED SHIP DATE AND ERROR CODE                                          000514  22900 11.11.11.164
 289 C                   Z-ADD     *ZERO         PEXPSH                                  000514  23000 11.11.11.164
                                                      0
 290 C                   MOVEL     *BLANKS       PERROR                                  000514  23100 11.11.11.164
 291 C                   Z-ADD     *ZEROS        KCUSNO                                  001002  23200 11.11.11.164
                                                      0
 292 C                   Z-ADD     *ZEROS        KSTORE                                  001002  23300 11.11.11.164
                                                      0
 293 C                   MOVEL     *BLANKS       KCUSNA                                  000323  23400 11.11.11.164
 294 C                   Z-ADD     *ZERO         EXPMDY                                  000323  23500 11.11.11.164
                                                      0
 295 C                   TIME                    TIMEN             6 0                   010501  23600 11.11.11.164
                                                111111
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.11.11.164 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111111
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.11.12.858 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111111
 297  * TEST F3                                                                          000323  23800 11.11.12.858
 298 C     *IN03         CABEQ     *ON           DONE                                    000323  23900 11.11.12.858
           0
 300 C     UDATE         CABEQ     090100        DONE                                    010113  24100 11.11.12.858
          120906
 302  *----------------------------------------------------------------                  000323  24300 11.11.12.858
 303  * VALIDATE ORDER # AND LINE #                                                      000323  24400 11.11.12.858
 304  *----------------------------------------------------------------                  000323  24500 11.11.12.858
 306  * GET ORDER DETAIL RECORD FOR ORDER# AND LINE#                                     990918  24700 11.11.12.858
 307 C                   Z-ADD     KORDER        OORDER                                  001007  24800 11.11.12.858
                                     1500
                                                   1500
 308 C                   Z-ADD     KLINE         OLINE                                   001007  24900 11.11.12.858
                                       2
                                                     2
 309  *---------------------------------------------------------------------             000909  25000 11.11.12.858
 310  **** ORDKEY        CHAIN     ODETREC                            25                 000909  25100 11.11.12.858
 311  **** *IN25         CABEQ     *ON           DISP01                   42             000909  25200 11.11.12.858
 313  * NO INDICATOR USED ON CHAIN  chain by file name, not record name                  030426  25400 11.11.12.858
 314  * AUDIT RPGIV CHAIN STATEMENT (NO ERROR INDICATOR)                                 000918  25500 11.11.12.858
 315 C     ordkey        chain     orderde                                         ph234 030504  25600 11.11.12.858
           000150000002
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430        ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317
ODEXPD-20070118 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
 316 C                   if        not%found                                   B01 ph234 030504  25700 11.11.12.858
 319 C                   END                                                   E01 ph235 030504  26000 11.11.12.858
 320  *---------------------------------------------------------------------             000909  26100 11.11.12.858
 322  * DID GET ORDER DETAIL RECORD                                                      000323  26300 11.11.12.858
 323  * CONVERT ODEXPD FORMAT YYYYMMDD TO PEXPSH FORMAT MMDDYY                           000317  26400 11.11.12.858
 324 C     odexpd        ifne      *zero                                       B01 ph543 030504  26500 11.11.12.858
         20070118
 325 C                   z-add     odexpd        expmd             4 0          01 ph543 030504  26600 11.11.12.858
                                 20070118
                                                   118
                                                                                                       11.11.12.858   PAGE
334 C                   eval                    answer = expmdy + expyy                 010522  27500 11.11.12.861
                                                  11814    11807       7
 335  *---------------------------------------------------------------------             010522  27600 11.11.12.861
 336  * AUDIT RPGIV eval starting in extended factor 2                                   010522  27700 11.11.12.861
 337 C                   eval      answer = expmdy + expyy                               010522  27800 11.11.12.861
                                    11814    11807       7
 338  *---------------------------------------------------------------------             000917  27900 11.11.12.862
 339 C                   EVAL      @STSC = *BLANKS                                       010730  28000 11.11.12.862
 342 C                   movel     '1'           @yes              1                     010730  28300 11.11.12.862
                                                 1
 343 C                   movel     '1'           @1stline          1                     010730  28400 11.11.12.862
                                                 1
 344 C                   setoff                                       33                 010730  28500 11.11.12.862
 345 C                   IF        @1stline = @yes and                         B01       010730  28600 11.11.12.862
                                   1          1
 346 C                             *IN33 = *OFF AND                            B01       050118  28700 11.11.12.862
                                   0
 347 C                             *INLR = *OFF                                B01       050118  28800 11.11.12.862
                                   0
 348 C                   movel     '2'           hold2             1            01       010730  28900 11.11.12.862
                                                 2
 349 C                   ENDIF                                             E01       010730  29000 11.11.12.862
 351 C                   dou       @1stline = @yes or                          B01 PDDD  010814  29200 11.11.12.862
 308 C                             *IN33 = *OFF                                B01 PDDD  010814  29300 11.11.12.862
 353 C                   movel     '3'           hold2             1            01       010730  29400 11.11.12.862
                                                 3
 354 C                   enddo                                             E01 PDDD  010814  29500 11.11.12.862
 355 C                   movel     '4'           hold2             1                     010730  29600 11.11.12.862
                                                 4
 356  * AUDIT RPGIV                                                                      010522  29700 11.11.12.862
 357 C                   eval      *IN50 = *ON                                           010522  29800 11.11.12.862
                                   1
 358 C                   eval                 *in55 = *OFF                               010710  29900 11.11.12.862
                                              0
 359 C                   eval                   *IN55 = *OFF                             010710  30000 11.11.12.862
                                                0
 360 C                   eval                     *in(60) = *ON                          010710  30100 11.11.12.862
                                                  1
 361 C                   eval                       *IN(60) = *ON                        010710  30200 11.11.12.862
                                                    1
 362 C                   eval                *IN = *OFF                                  010710  30300 11.11.12.862

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 363  *---------------------------------------------------------------------             010522  30400 11.11.12.862
 364  * AUDIT RPGIV DOW STATEMENT                                                        000918  30500 11.11.12.862
 365 C                   Z-ADD     *ZERO         COUNTER           2 0                   000918  30600 11.11.12.862
                                                       0
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.12.862
                                                       0
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.12.862
                                                       1
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.12.862
                                                       1
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.12.862
                                                       2
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.12.862
                                                       2
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.12.862
                                                       3
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.12.862
                                                       3
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.12.862
                                                       4
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.12.862
                                                       4
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.12.862
                                                       5
                                                                                                       11.10.48.733     PAGE
366 C                   DOW                     COUNTER < 6                    B01       000918  30700 11.11.12.862
                                                       5
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.12.862
                                                       6
 368 C                   ENDDO                                                 E01       000918  30900 11.11.12.862
 369  *---------------------------------------------------------------------             000918  31000 11.11.12.862
 370  * AUDIT RPGIV select, WHEN, OTHER STATEMENTS                                       001029  31100 11.11.12.862
 371 C                   SELECT                                                B01       001029  31200 11.11.12.862
 372 C                   WHEN      COUNTER = 6                                 X01       001029  31300 11.11.12.862
                                         6
 372 C                   WHEN      COUNTER = 6                                 X01       001029  31300 11.11.12.862
                                         6
 373 C                   Z-ADD     *ZERO         COUNTER                        01       001029  31400 11.11.12.862
                                                       0
 375 C                   ENDSL                                                 E01       001029  31600 11.11.12.863
 376  * AUDIT RPGIV select, WHEN, OTHER STATEMENTS                                       010912  31700 11.11.12.863
 377 C                   movel     'P'           @mode             1                     010912  31800 11.11.12.863
                                                 P
 378 C                   z-add     1             PHSCNO            1 0                   010912  31900 11.11.12.863
                                                      1
 379 C                   SELECT                                                B01       010912  32000 11.11.12.864
 380 C                   WHEN      @mode = 'P'                                 X01       010912  32100 11.11.12.864
                                   P
 381 C                              OR PHSCNO = 1                              X01       010912  32200 11.11.12.864
                                            1
 380 C                   WHEN      @mode = 'P'                                 X01       010912  32100 11.11.12.864
                                   P
 337 C                              OR PHSCNO = 1                              X01       010912  32200 11.11.12.864
                                            1
 382 C                   Z-ADD     *ZERO         COUNTER                        01       010912  32300 11.11.12.864
                                                       0
 385 C                   ENDSL                                                 E01       010912  32600 11.11.12.864
 386  *---------------------------------------------------------------------             010912  32700 11.11.12.864
 387 C                   if        counter <> 0                                B01       010912  32800 11.11.12.864
                                         0
 389 C                   endif                                                 E01       010912  33000 11.11.12.864
 390 C                   movel     'R'           @mode             1                     010912  33100 11.11.12.864
                                                 R
 391 C                   movel     '3'           RTYP              1                     010912  33200 11.11.12.864
                                                 3
 392 C                   SELECT                                                B01       010912  33300 11.11.12.864
 393 C                   WHEN      @mode = 'R'                                 X01       010912  33400 11.11.12.864
                                   R
 394 C                             AND RTYP = '3'                              X01       010912  33500 11.11.12.864
                                       3
 393 C                   WHEN      @mode = 'R'                                 X01       010912  33400 11.11.12.864
                                   R
 350 C                             AND RTYP = '3'                              X01       010912  33500 11.11.12.864
                                       3
 395 C                   Z-ADD     *ZERO         COUNTER                        01       010912  33600 11.11.12.864
                                                       0
 398 C                   ENDSL                                                 E01       010912  33900 11.11.12.864
 399  *---------------------------------------------------------------------             001029  34000 11.11.12.864
 400  * AUDIT RPGIV IF STATEMENT                                                         001029  34100 11.11.12.864
 401 C                   IF        COUNTER = 0                                 B01       001029  34200 11.11.12.864
                                         0
 402 C                   Z-ADD     3             COUNTER                        01       001029  34300 11.11.12.864
```

```
                                                      3
403 C                   ENDIF                                                 E01       001029  34400 11.11.12.864
404  *----------------------------------------------------------------------            001029  34500 11.11.12.864
405 C                   if        counter > 0                                 B01       010522  34600 11.11.12.865
                                        3
406 C                   endif                                                 E01       010522  34700 11.11.12.865
407  *----------------------------------------------------------------------            010522  34800 11.11.12.865
408 C                   if        counter <> 0                                B01       010522  34900 11.11.12.865
                                        3
409 C                   endif                                                 E01       010522  35000 11.11.12.865
410  *----------------------------------------------------------------------            010522  35100 11.11.12.866
411  * AUDIT RPGIV EXTENDED FACTOR 2 CONDITIONAL AND/OR COMPLEX STATEMENTS              010602  35200 11.11.12.866
412 C                   Z-ADD     2             COUNTER                                 010602  35300 11.11.12.866
                                                      2
413 C                   Z-ADD     7             answer                                  010602  35400 11.11.12.866
                                                     7
414 C                   Z-ADD     *zero         final             7 2                   010607  35500 11.11.12.866
                                                  .00
415 C                   Z-sub     *zero         sum               6 1                   010704  35600 11.11.12.866
                                                 .0
416 C                   Z-ADD     3115          total             8 0                   010604  35700 11.11.12.866
                                                 3115
                                                                                                      11.10.48.733     PAGE
417 C                   Z-ADD     112           net               3 0                   010607  35800 11.11.12.866
                                                112
418 C                   IF        COUNTER = 0                                 B01       010602  35900 11.11.12.866
                                        2
419 C                             OR COUNTER = 2                              B01       010602  36000 11.11.12.866
                                           2
420 C                             OR COUNTER = 4                              B01       010602  36100 11.11.12.866
                                           2
421 C                             AND ANSWER = 7                              B01       010602  36200 11.11.12.866
                                           7
446 C                                     CCCC - DIFFERENCE +160 + EXTRA +              010617  38700 11.11.12.866
                                          7.12      1444.20           87
447 C                                     188  - INTERIM + EXTRA2                       010617  38800 11.11.12.866
                                                 1229.85      105
448 C                                   - 33.15 + GROSS                                 010617  38900 11.11.12.866
                                                 163.23
449 C                                   + MORE                                          010617  39000 11.11.12.866
                                         17.00
452 C                             + 44 - GROSS - EXTRA2                                 010617  39300 11.11.12.866
                                        163.23      105
422 C                   Z-ADD     3             COUNTER                        01       010604  36300 11.11.12.866
                                                      3
423 C                   ENDIF                                                 E01       010602  36400 11.11.12.866
424 C     sum           add       total         final                                   010604  36500 11.11.12.866
            .0                     3115
                                              3115.00
425 C                   eval      final = answer + counter + 5                          010602  36600 11.11.12.866
                                  15.00        7         3
426 C                   eval      sum = 4 + 6 - 2 + 1555                                010602  36700 11.11.12.866
                               1563.0
427 C                   eval      final = answer * counter                              010610  36800 11.11.12.866
                                  21.00        7         3
428 C                   eval      final = answer* counter                               010610  36900 11.11.12.866
                                  21.00        7        3
429 C                   eval      final = answer *counter                               010610  37000 11.11.12.866
                                  21.00        7        3
430 C                   eval      final = answer*counter                                010610  37100 11.11.12.868
                                  21.00        7       3
431 C                   eval      final = answer / counter                              010610  37200 11.11.12.868
                                   2.33        7         3
432 C                   eval      final = answer/counter                                010610  37300 11.11.12.868
                                   2.33        7       3
433 C                   eval      sum = 4 + 6 - counter + 1555                          010605  37400 11.11.12.868
                               1562.0                 3
434 C                   eval        total = 4 + 6 + sum                                 010603  37500 11.11.12.868
                                     1572        1562.0
435 C                   eval                    sum = 4 + 6                             010603  37600 11.11.12.868
                                               10.0
436 C                   eval      total = 4 + sum + 6 + answer + final - net            010606  37700 11.11.12.869
```

```
                                     82-     10.0            7
                                                                  2.33   112
437  *----------------------------------------------------------------------            010602  37800 11.11.12.869
438 C                   Z-ADD     7.12          cccc              5 2                   010617  37900 11.11.12.869
                                                7.12
439 C                   Z-ADD     163.23        gross             6 2                   010617  38000 11.11.12.869
                                               163.23
440 C                   z-add     1444.2        difference        8 2                   010617  38100 11.11.12.869
                                                   1444.20
441 C                   z-add     43.8          interim           8 2                   010617  38200 11.11.12.869
                                                  43.80
442 C                   z-add     87            extra             5 0                   010617  38300 11.11.12.869
                                                   87
443 C                   z-add     105           extra2            5 0                   010617  38400 11.11.12.869
                                                   105
444 C                   z-add     17            more              8 2                   010617  38500 11.11.12.869
                                               17.00
445 C                   eval      final = answer + counter +                            010617  38600 11.11.12.869
                                 783.80-       7         3
402 C                                     CCCC - DIFFERENCE +160 + EXTRA +              010617  39000 11.11.12.869
                                          7.12      1444.20           87
403 C                                     188  - INTERIM + EXTRA2                       010617  39000 11.11.12.869
                                                   43.80      105
404 C                                   - 33.15 + GROSS                                 010617  39000 11.11.12.869
                                                 163.23
405 C                                   + MORE                                          010617  39000 11.11.12.869
                                         17.00
450 C                   eval      net  = counter + cccc                                 010617  39100 11.11.12.869
                                   10          3   7.12
451 C                   eval      interim = answer - cccc + net + difference            010617  39200 11.11.12.869
                                  1229.85        7   7.12    10      1444.20
408 C                             + 44 - GROSS - EXTRA2                                 010617  39300 11.11.12.869
                                        163.23      105
                                                                                                      11.10.48.733   PAGE
453  *----------------------------------------------------------------------            010617  39400 11.11.12.869
454 C                   movel     '2'           movsw             1                     010604  39500 11.11.12.869
                                                2
455  * AUDIT RPGIV IF STATEMENT with alpha extended factor 2                            010604  39600 11.11.12.869
456 C                   if        movsw = '2'                                 B01       010604  39700 11.11.12.869
                                  2
457 C                             OR MOVSW = '3'                              B01       010604  39800 11.11.12.869
                                     2
458 C                   movel     '1'           movsw             1            01       010604  39900 11.11.12.869
                                                1
459 C                   endif                                                 E01       010604  40000 11.11.12.869
460 C                   movel     '5'           movsw             1                     010604  40100 11.11.12.869
                                                5
461 C                   if        movsw = '5'                                 B01       010604  40200 11.11.12.869
                                  5
462 C                   endif                                                 E01       010604  40300 11.11.12.869
463  *----------------------------------------------------------------------            010604  40400 11.11.12.869
464 C                   Z-ADD     2             COUNTER                                 010615  40500 11.11.12.869
                                                      2
465 C                   Z-ADD     7             answer                                  010615  40600 11.11.12.869
                                                     7
466 C                   Z-ADD     14.2          gggggg            6 2                   010615  40700 11.11.12.869
                                                 14.20
467 C                   IF        COUNTER = 0 and  gggggg =5 and              B01       010615  40800 11.11.12.869
                                        2           14.20
468 C                                COUNTER > 5 AND                          B01       010615  40900 11.11.12.869
                                           2
469 C                                COUNTER < 3 OR ANSWER = 6 AND            B01       010615  41000 11.11.12.869
                                           2             7
470 C                                COUNTER < 2                              B01       010615  41100 11.11.12.869
                                           2
471 C                             OR COUNTER = 2                              B01       010615  41200 11.11.12.869
                                           2
472 C                             AND ANSWER = 7 OR                           B01       010615  41300 11.11.12.869
                                           7
473 C                             ANSWER = 5                                  B01       010615  41400 11.11.12.869
                                       7
474 C                   Z-ADD     3             COUNTER                        01       010615  41500 11.11.12.869
```

```
                                                       3
 475 C                   ENDIF                                                 E01       010615  41600 11.11.12.869
 476  *                                                                -----             010615  41700 11.11.12.869
 477  *----------------------------------------------------------------------            010615  41800 11.11.12.869
 478  * AUDIT RPGIV MOVEL PADDED WITH BLANKS STATEMENT                                   010522  41900 11.11.12.869
 479 C                   MOVEL     *ALL'M'       TESTML           20                     001029  42000 11.11.12.869
                                                 MMMMMMMMMMMMMMMMMMMM
 480 C                   MOVEL(P)  'LEFT'        TESTML                                  001029  42100 11.11.12.869
                                                 LEFT
 481  *----------------------------------------------------------------------            010330  42200 11.11.12.869
 482  * all lower case statements                                                        010330  42300 11.11.12.869
 483 C                   movel     *ALL'L'       testlo           20                     010330  42400 11.11.12.870
                                                 LLLLLLLLLLLLLLLLLLLL
 484 C                   z-add     11111         aaaaaaaaaa        6 0                   010330  42500 11.11.12.871
                                                      11111
 485 C                   z-add     2222222       bbbbbbbbbb        8 3                   010618  42600 11.11.12.871
                                                  22222.000
 486 C     aaaaaaaaaa    add       bbbbbbbbbb    cccccccccc        8 0                   010330  42700 11.11.12.871
                11111               22222.000
                                                      33333
 487  *----------------------------------------------------------------------            001029  42800 11.11.12.871
 488  * AUDIT RPGIV TIME STATEMENT                                                       001029  42900 11.11.12.871
 489 C                   TIME                    TIMENOW           6 0                   001029  43000 11.11.12.871
                                                  111112
 490  *----------------------------------------------------------------------            001029  43100 11.11.12.871
 491 C                   movel     '2'           movsw            1                      010530  43200 11.11.12.871
                                                 2
 492  * AUDIT RPGIV IF STATEMENT                                                         010113  43300 11.11.12.871
 493 C                   if        movsw = *blanks                             B01       010113  43400 11.11.12.871
                                   2
 494 C                             OR COUNTER = 0                              B01       010113  43500 11.11.12.871
                                            3
 496 C                   endif                                                 E01       010113  43700 11.11.12.871
 497  *----------------------------------------------------------------------            010113  43800 11.11.12.871
 499 C     100           DIV       5.25          NET               3 0        01         000323  48400 11.11.12.872
                                                  19
 500 C                   MVR                     FRACT             4 4        01         000323  48500 11.11.12.872
                                                 .2500
 502  *  GOT ORDER DETAIL, PRINT IT                                                      000514  44300 11.11.12.872
 503 C                   EXCEPT    PRTDET                                                000514  44400 11.11.12.872
 505  * GET THE CUSTOMER MASTER STORE RECORD                                             991225  44600 11.11.12.872
                                                                                                       11.10.48.733   PAGE
 506 C                   Z-ADD     ODCUST        CUCUST                                  991225  44700 11.11.12.872
                                     1000
                                                   1000
 507 C                   Z-ADD     ODSTOR        CUSTOR                                  991225  44800 11.11.12.872
                                        1
                                                      1
 508  *-----------------------------------------------------------------                 010118  44900 11.11.12.872
 509  * THIS ROUTINE CAUSES THE WRONG CUSTOMER TO BE DISPLAYED                           010118  45000 11.11.12.872
 510  * IF THE ORDER DETAIL FILE HAS BEEN UPDATED TWICE, AND ONLY TWICE                  010118  45100 11.11.12.872
 511  * FIND THE ERROR ON THE AUDIT REPORT BY SCANNING FOR 2050                          010118  45200 11.11.12.872
 512 C     UPDREC        IFEQ      2                                           B01       010118  45300 11.11.12.872
                1
 521 C                   ENDIF                                                 E01       010118  46200 11.11.12.872
 523  *-----------------------------------------------------------------                 010118  46400 11.11.12.872
 524 C     CUSKEY        CHAIN     CUSTREC1                   30                         000717  46500 11.11.12.872
       N30 00010000000001
CUCUST-0001000  CUSTOR-0000001  CUNAME-ABC  STORES  INC                        CUAD1-423  MONTGOMERY  AVENUE          CUAD2-
CUSTA-PA
 525 C                   z-add     *all'1'       aa                3 0                   050102  46600 11.11.12.872
                                                111
 526 C                   z-add     *all'2'       bb                3 0                   050102  46700 11.11.12.872
                                                222
 527 C                   z-add     *all'3'       cc                3 0                   050102  46800 11.11.12.872
                                                333
 528 C                   z-add     *all'4'       dd                3 0                   050102  46900 11.11.12.872
                                                444
 529 C                   z-add     *all'5'       ee                3 0                   050102  47000 11.11.12.872
                                                555
 530 C                   z-add     *all'6'       ff                3 0                   050102  47100 11.11.12.872
                                                666
 531 C                   z-add     *all'7'       gg                3 0                   050102  47200 11.11.12.872
                                                777
 532 C                   z-add     *all'8'       hh                3 0                   050102  47300 11.11.12.876
                                                888
 533 C                   z-add     *all'9'       ii                3 0                   050102  47400 11.11.12.876
                                                999
 534 C                   z-add     *zeros        total             8 0                   050102  47500 11.11.12.876
                                                     0
 536   total = aa + bb + cc + dd + ee + ff + gg + hh + ii;                               050102  47700 11.11.12.876
        4995  111  222  333  444  555  666  777  888  999
 538 c                   eval      total = aa + bb + cc + dd + ee + ff + gg + hh         050102  47900 11.11.12.876
                                    3996  111  222  333  444  555  666  777  888
 539 C     *IN30         IFEQ      *OFF                                        B01       991225  48000 11.11.12.876
           0
 540  * GOT CUSTOMER MASTER                                                              991225  48100 11.11.12.876
 541 C                   Z-ADD     CUCUST        KCUSNO                         01       001007  48200 11.11.12.876
                                     1000
                                                   1000
 542 C                   Z-ADD     CUSTOR        KSTORE                         01       010118  48300 11.11.12.876
                                        1
                                                      1
 543 C                   MOVEL     CUNAME        KCUSNA                         01       000323  48400 11.11.12.876
                                   ABC STORES INC
                                                 ABC STORES INC
 544 C                   MOVEL     CUNAME        PCUSNA                         01       000323  48500 11.11.12.876
                                   ABC STORES INC
                                                 ABC STORES INC
 545 C                   EXCEPT    PRTCUS                                       01       000323  48600 11.11.12.876
 548 C                   ENDIF                                         E01       991225  48900 11.11.12.876
 550  * DISPLAY DETAIL SCREEN                                                            000323  49100 11.11.12.876
 551 C     DISP02        TAG                                                             000514  49200 11.11.12.876
 552 C                   TIME                    TIMEN             6 0                   010501  49300 11.11.12.876
                                                111112
 553 C                   EXFMT     NEWEXPD2                                              051007  49400 11.11.12.876 WRITE
*IN03-0 *IN43-0 EXPMDY-011807 KCUSNO-0001000 KCUSNA-ABC STORES INC                  UDATE-120906 KSTORE-0000001 TIMEN-111112
KORDER-0001500 KLINE-00002
 553 C                   EXFMT     NEWEXPD2                                              051007  49400 11.11.13.678 READ
*IN03-0 *IN43-0 EXPMDY-011807 KCUSNO-0001000 KCUSNA-ABC STORES INC                  UDATE-120906 KSTORE-0000001 TIMEN-111112
KORDER-0001500 KLINE-00002
 554  * TEST F3                                                                          000323  49500 11.11.13.681
 555 C     *IN03         CABEQ     *ON           DONE                                    000323  49600 11.11.13.681
           0
 557  * VALIDATE CHANGED DATE, AND UPDATE ORDER DETAIL                                   000323  49800 11.11.13.681
 559  * CONVERT EXPMDY FORMAT MMDDYY TO ODEXPD FORMAT YYYYMMDDYY                         000323  50000 11.11.13.681
 561 C                   Z-ADD     EXPMDY        YY                2 0          YY       000323  50200 11.11.13.681
                                    11807
                                                  7
 562 C     EXPMDY        DIV       100           MMDD              4 0          MMDD     000323  54700 11.11.13.681
            11807                                 118
 563 C     YY            MULT      10000         Y4MMDD            8 0          00YY0000 000323  50400 11.11.13.681
            7                                     70000
 564 C                   ADD       MMDD          Y4MMDD                         00YYMMDD 000323  50500 11.11.13.681
                                    118
                                                  70118
                                                                                                       11.10.48.733   PAGE
 565 C     YY            IFGT      40                                          B01       000323  50600 11.11.13.681
            7
 567 C                   ELSE                                                  X01       000323  50800 11.11.13.681
 568 C                   ADD       20000000      Y4MMDD                         01       000323  50900 11.11.13.681
                                               20070118
 569 C                   END                                                   E01       000323  51000 11.11.13.681
 570 C                   Z-ADD     Y4MMDD        ODEXPD                                  000323  51100 11.11.13.681
                                 20070118
                                               20070118
 571  * COMPLEX IF STATEMENT                                                             010614  51200 11.11.13.681
 572 C                   IF        COUNTER = 0 and                             B01       010614  51300 11.11.13.681
                                         3
 573 C                                COUNTER > 5 AND                          B01       010614  51400 11.11.13.681
                                            3
 574 C                                COUNTER < 3 OR ANSWER = 6 AND            B01       010614  51500 11.11.13.681
                                            3             7
 575 C                                COUNTER < 2                              B01       010614  51600 11.11.13.681
```

```
                                            3
 576 C                             OR COUNTER = 2                              B01       010614  51700 11.11.13.681
                                            3
 577 C                             AND ANSWER = 7 OR                           B01       010614  51800 11.11.13.681
                                            7
 578 C                             ANSWER = 5                                  B01       010614  51900 11.11.13.681
                                        7
 580 C                   ENDIF                                                 E01       010614  52100 11.11.13.681
 582  * TEST FOR FIELD OVER 100...                                                       010411  52300 11.11.13.681
 583 C                   MOVE      *ALL'#'       ALL#            256                     010411  52400 11.11.13.681
VAR     ALL#                                                                                      1     -                100
####################################################################################################
VAR      ALL#                                                                                101      -                 200
####################################################################################################
VAR ALL#              201 -   256  ########################################################
 584 C                   MOVE      ALL#          ALL$            256                     010411  52500 11.11.13.681
VAR     ALL#                                                                                      1     -                100
####################################################################################################
VAR      ALL#                                                                                101      -                 200
####################################################################################################
VAR ALL#              201 -   256  ########################################################
VAR     ALL$                                                                                      1     -                100
####################################################################################################
VAR      ALL$                                                                                101      -                 200
####################################################################################################
VAR ALL$              201 -   256  ########################################################
 586 C                   MOVE      '2'           MOVSW                                   010429  52700 11.11.13.681
                                                 2
 587 C                   MOVE      '3'           MOVSW1            1                     010429  52800 11.11.13.681
                                                 3
 588 C                   MOVE      '4'           MOVSW2            1                     010429  52900 11.11.13.681
                                                 4
 589 C                   MOVE      '5'           MOVSW3            1                     010429  53000 11.11.13.681
                                                 5
 590 C                   MOVE      '5'           MOVSW4            1                     010429  53100 11.11.13.681
                                                 5
 591 C                   MOVE      'A'           HLD1              1                     010429  53200 11.11.13.681
                                                 A
 593 C     EXPMDY        IFEQ      UDATE                                       B01       010420  53400 11.11.13.681
            11807                  120906
 594 C     MOVSW         ANDEQ     *BLANK                                       01       000514  53500 11.11.13.681
           2
 595 C     MOVSW         OREQ      '5'                                          01       010429  53600 11.11.13.681
           2
 596 C     MOVSW1        OREQ      MOVSW2                                       01       010429  53700 11.11.13.681
           3                       4
 597 C     MOVSW3        ORNE      MOVSW4                                       01       010429  53800 11.11.13.681
           5                       5
 598 C     MOVSW         OREQ      '8'                                          01       010429  53900 11.11.13.681
           2
 599 C     MOVSW         OREQ      '9'                                          01       010429  54000 11.11.13.681
           2
 600 C     MOVSW         OREQ      'C'                                          01       010429  54100 11.11.13.681
           2
 601 C     MOVSW         OREQ      'D'                                          01       010429  54200 11.11.13.681
           2
 602 C     MOVSW         OREQ      HLD1                                         01       010429  54300 11.11.13.681
           2                       A
 603 C     MOVSW3        ANDNE     MOVSW2                                       01       010429  54400 11.11.13.681
           5                       4
 605 C                   END                                               E01       000323  54600 11.11.13.681
 607  * UPDATE ORDER DETAIL EXPECTED SHIP DATE                                           000323  54800 11.11.13.681
 608 C                   ADD       1             UPDREC            6 0                   010118  54900 11.11.13.681
                                                      2
 609 C                   UPDATE    ODETREC                                               000323  55000 11.11.13.681
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430         ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317
ODEXPD-20070118 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
 611  * WRITE UPDATED ORDER DETAIL TO A WORK FILE                                        000402  55200 11.11.13.683
                                                                                                       11.10.48.733   PAGE
 612 C                   CLEAR                   ODETWRK                                 000402  55300 11.11.13.683
 613 C                   Z-ADD     ODORD#        WDORD#                                  000402  55400 11.11.13.683
                                     1500
```

```
                                                   1500
 614 C                   Z-ADD     ODLINE        WDLINE                                  000402  55500 11.11.13.683
                                        2
                                                      2
 615 C                   Z-ADD     ODCUST        WDCUST                                  000402  55600 11.11.13.683
                                     1000
                                                   1000
 616 C                   Z-ADD     ODSTOR        WDSTOR                                  000402  55700 11.11.13.683
                                        1
                                                      1
 617 C                   MOVEL     ODITEM        WDITEM                                  000402  55800 11.11.13.683
                                   Y2430
                                                 Y2430
 618 C                   Z-ADD     ODPRIC        WDPRIC                                  000402  55900 11.11.13.683
                                    25.15
                                                  25.15
 619 C                   Z-ADD     ODQTY         WDQTY                                   000402  56000 11.11.13.683
                                        3
                                                      3
 620 C                   Z-ADD     ODREQD        WDREQD                                  000402  56100 11.11.13.683
                                 20000317
                                               20000317
 621 C                   Z-ADD     ODEXPD        WDEXPD                                  000402  56200 11.11.13.683
                                 20070118
                                               20070118
 622 C                   Z-ADD     ODSHPD        WDSHPD                                  000402  56300 11.11.13.683
                                        0
                                                      0
 623 C                   Z-ADD     ODINV#        WDINV#                                  000402  56400 11.11.13.683
                                        0
                                                      0
 624 C                   MOVEL     ODSTAT        WDSTAT                                  000402  56500 11.11.13.683
                                   O
                                                 O
 625 C                   MOVEL     ODX           WDX                                     000402  56600 11.11.13.683
 627  * WRITE ORDERWK                                                                    000402  56800 11.11.13.683
 628 C                   WRITE     ODETWRK                                               000402  56900 11.11.13.683
WDORD#-0001500 WDLINE-00002 WDCUST-0001000 WDSTOR-0000001 WDITEM-Y2430                  WDPRIC-0002515 WDQTY-0000003 WDREQD-
20000317 WDEXPD-20070118 WDSHPD-00000000 WDINV#-0000000 WDSTAT-O WDX-
 630  * NESTED IF STATEMENTS WITH AND/OR                                                 010429  57100 11.11.13.683
 631 C     aaaaaaaaaa    ifeq      bbbbbbbbbb                                  B01       010429  57200 11.11.13.683
                11111               22222.000
 632 C     UPDREC        andeq     YY                                           01       010429  57300 11.11.13.683
                2                   7
 637 C                   END                                                   E01       010429  57800 11.11.13.684
 639  * REDISPLAY FIRST SCREEN                                                           000323  58000 11.11.13.684
 640 C                   GOTO      DISP01                                                000514  58100 11.11.13.684
 287 C     DISP01        TAG                                                             000514  22800 11.11.13.684
 288  * CLEAR EXPECTED SHIP DATE AND ERROR CODE                                          000514  22900 11.11.13.684
 289 C                   Z-ADD     *ZERO         PEXPSH                                  000514  23000 11.11.13.684
                                                      0
 290 C                   MOVEL     *BLANKS       PERROR                                  000514  23100 11.11.13.684
 291 C                   Z-ADD     *ZEROS        KCUSNO                                  001002  23200 11.11.13.684
                                                      0
 292 C                   Z-ADD     *ZEROS        KSTORE                                  001002  23300 11.11.13.684
                                                      0
 293 C                   MOVEL     *BLANKS       KCUSNA                                  000323  23400 11.11.13.684
 294 C                   Z-ADD     *ZERO         EXPMDY                                  000323  23500 11.11.13.684
                                                      0
 295 C                   TIME                    TIMEN             6 0                   010501  23600 11.11.13.684
                                                111113
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.11.13.684 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111113
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.11.14.058 READ
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111113
 297  * TEST F3                                                                          000323  23800 11.11.14.058
 298 C     *IN03         CABEQ     *ON           DONE                                    000323  23900 11.11.14.058
           0
 300 C     UDATE         CABEQ     090100        DONE                                    010113  24100 11.11.14.058
          120906
 302  *-----------------------------------------------------------------                 000323  24300 11.11.14.058
 303  * VALIDATE ORDER # AND LINE #                                                      000323  24400 11.11.14.058
```

```
                                                                                                       11.10.48.733   PAGE
 304  *-----------------------------------------------------------------                 000323  24500 11.11.14.058
 306  * GET ORDER DETAIL RECORD FOR ORDER# AND LINE#                                     990918  24700 11.11.14.058
 307 C                   Z-ADD     KORDER        OORDER                                  001007  24800 11.11.14.058
                                     1500
                                                   1500
 308 C                   Z-ADD     KLINE         OLINE                                   001007  24900 11.11.14.058
                                       2
                                                     2
 309  *----------------------------------------------------------------------            000909  25000 11.11.14.058
 310  **** ORDKEY        CHAIN     ODETREC                           25        25 IS    000909  25100 11.11.14.058
 311  **** *IN25         CABEQ     *ON           DISP01                  42             000909  25200 11.11.14.058
 313  * NO INDICATOR USED ON CHAIN  chain by file name, not record name                   030426  25400 11.11.14.058
 314  * AUDIT RPGIV CHAIN STATEMENT (NO ERROR INDICATOR)                                  000918  25500 11.11.14.058
 315 C     ordkey        chain     orderde                                         ph234 030504  25600 11.11.14.058
           000150000002
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430        ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317
ODEXPD-20070118 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
 316 C                   if        not%found                                   B01 ph234 030504  25700 11.11.14.058
 319 C                   END                                                   E01 ph235 030504  26000 11.11.14.060
 320  *----------------------------------------------------------------------            000909  26100 11.11.14.060
 322  * DID GET ORDER DETAIL RECORD                                                      000323  26300 11.11.14.060
 323  * CONVERT ODEXPD FORMAT YYYYMMDD TO PEXPSH FORMAT MMDDYY                           000317  26400 11.11.14.060
 324 C     odexpd        ifne      *zero                                       B01 ph543 030504  26500 11.11.14.060
         20070118
 325 C                   z-add     odexpd        expmd             4 0          01 ph543 030504  26600 11.11.14.060
                                 20070118
                                                   118
 326 C     odexpd        DIV       10000         expyy             2 0          YY       001029  31100 11.11.14.060
         20070118                                    7
 327 C     expmd         mult      100           expmdy            6 0          01       010330  26800 11.11.14.060
             118                                  11800
 328 C     expmdy        add       expyy         expmdy                         01       010330  26900 11.11.14.060
            11807                      7
                                                  11807
 329 C                   endif                                                 E01       010330  27000 11.11.14.060
 331  *----------------------------------------------------------------------            000917  27200 11.11.14.060
 332  * AUDIT RPGIV EVAL STATEMENT (LOWER CASE)                                          010113  27300 11.11.14.060
 333 C                   z-add     *zero         answer            7 0          EVAL RESU010113  27400 11.11.14.060
                                                      0
 334 C                   eval                    answer = expmdy + expyy                 010522  27500 11.11.14.060
                                                  11814    11807       7
 335  *----------------------------------------------------------------------            010522  27600 11.11.14.060
 336  * AUDIT RPGIV eval starting in extended factor 2                                   010522  27700 11.11.14.060
 337 C                   eval      answer = expmdy + expyy                               010522  27800 11.11.14.060
                                    11814    11807       7
 338  *----------------------------------------------------------------------            000917  27900 11.11.14.060
 339 C                   EVAL      @STSC = *BLANKS                                       010730  28000 11.11.14.060
 342 C                   movel     '1'           @yes              1                     010730  28300 11.11.14.060
                                                 1
 343 C                   movel     '1'           @1stline          1                     010730  28400 11.11.14.060
                                                 1
 344 C                   setoff                                       33                 010730  28500 11.11.14.061
 345 C                   IF        @1stline = @yes and                         B01       010730  28600 11.11.14.061
                                   1          1
 346 C                             *IN33 = *OFF AND                            B01       050118  28700 11.11.14.061
                                   0
 347 C                             *INLR = *OFF                                B01       050118  28800 11.11.14.061
                                   0
 348 C                   movel     '2'           hold2             1            01       010730  28900 11.11.14.061
                                                 2
 349 C                   ENDIF                                                 E01       010730  29000 11.11.14.061
 351 C                   dou       @1stline = @yes or                          B01 PDDD  010814  29200 11.11.14.061
 308 C                             *IN33 = *OFF                                B01 PDDD  010814  29300 11.11.14.061
 353 C                   movel     '3'           hold2             1            01       010730  29400 11.11.14.061
                                                 3
 354 C                   enddo                                                 E01 PDDD  010814  29500 11.11.14.061
 355 C                   movel     '4'           hold2             1                     010730  29600 11.11.14.061
                                                 4
 356  * AUDIT RPGIV                                                                      010522  29700 11.11.14.061
 357 C                   eval      *IN50 = *ON                                           010522  29800 11.11.14.061
                                   1
 358 C                   eval                 *in55 = *OFF                               010710  29900 11.11.14.061
                                              0
 359 C                   eval                   *IN55 = *OFF                             010710  30000 11.11.14.061
                                                0
 360 C                   eval                     *in(60) = *ON                          010710  30100 11.11.14.061
                                                  1
 361 C                   eval                       *IN(60) = *ON                        010710  30200 11.11.14.061
                                                    1
                                                                                                       11.10.48.733   PAGE
 362 C                   eval                *IN = *OFF                                  010710  30300 11.11.14.061

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 363  *----------------------------------------------------------------------            010522  30400 11.11.14.061
 364  * AUDIT RPGIV DOW STATEMENT                                                        000918  30500 11.11.14.061
 365 C                   Z-ADD     *ZERO         COUNTER           2 0                   000918  30600 11.11.14.061
                                                       0
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.14.061
                                                       0
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.14.061
                                                       1
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.14.061
                                                       1
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.14.061
                                                       2
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.14.061
                                                       2
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.14.061
                                                       3
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.14.061
                                                       3
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.14.061
                                                       4
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.14.061
                                                       4
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.14.061
                                                       5
 366 C                   DOW                     COUNTER < 6                   B01       000918  30700 11.11.14.061
                                                       5
 367 C                   ADD       1             COUNTER                        01       000918  30800 11.11.14.061
                                                       6
 368 C                   ENDDO                                                 E01       000918  30900 11.11.14.061
 369  *----------------------------------------------------------------------            000918  31000 11.11.14.063
 370  * AUDIT RPGIV select, WHEN, OTHER STATEMENTS                                       001029  31100 11.11.14.063
 371 C                   SELECT                                                B01       001029  31200 11.11.14.063
 372 C                   WHEN      COUNTER = 6                                 X01       001029  31300 11.11.14.063
                                         6
 372 C                   WHEN      COUNTER = 6                                 X01       001029  31300 11.11.14.063
                                         6
 373 C                   Z-ADD     *ZERO         COUNTER                        01       001029  31400 11.11.14.063
                                                       0
 375 C                   ENDSL                                                 E01       001029  31600 11.11.14.063
 376  * AUDIT RPGIV select, WHEN, OTHER STATEMENTS                                       010912  31700 11.11.14.063
 377 C                   movel     'P'           @mode             1                     010912  31800 11.11.14.063
                                                 P
 378 C                   z-add     1             PHSCNO            1 0                   010912  31900 11.11.14.063
                                                      1
 379 C                   SELECT                                                B01       010912  32000 11.11.14.063
 380 C                   WHEN      @mode = 'P'                                 X01       010912  32100 11.11.14.063
                                   P
 381 C                              OR PHSCNO = 1                              X01       010912  32200 11.11.14.063
                                            1
 380 C                   WHEN      @mode = 'P'                                 X01       010912  32100 11.11.14.063
                                   P
 337 C                              OR PHSCNO = 1                              X01       010912  32200 11.11.14.063
                                            1
 382 C                   Z-ADD     *ZERO         COUNTER                        01       010912  32300 11.11.14.063
                                                       0
 385 C                   ENDSL                                                 E01       010912  32600 11.11.14.063
 386  *----------------------------------------------------------------------            010912  32700 11.11.14.063
 387 C                   if        counter <> 0                                B01       010912  32800 11.11.14.063
                                         0
 389 C                   endif                                                 E01       010912  33000 11.11.14.063
```

```
390 C                   movel     'R'           @mode             1                     010912  33100 11.11.14.063
                                                R
391 C                   movel     '3'           RTYP              1                     010912  33200 11.11.14.063
                                                3
392 C                   SELECT                                                B01       010912  33300 11.11.14.063
393 C                   WHEN      @mode = 'R'                                 X01       010912  33400 11.11.14.063
                                  R
394 C                             AND RTYP = '3'                              X01       010912  33500 11.11.14.063
                                      3
393 C                   WHEN      @mode = 'R'                                 X01       010912  33400 11.11.14.063
                                  R
350 C                             AND RTYP = '3'                              X01       010912  33500 11.11.14.063
                                      3
395 C                   Z-ADD     *ZERO         COUNTER                        01       010912  33600 11.11.14.063
                                                      0
398 C                   ENDSL                                                 E01       010912  33900 11.11.14.063
399  *----------------------------------------------------------------------            001029  34000 11.11.14.063
                                                                                                      11.10.48.733   PAGE
400  * AUDIT RPGIV IF STATEMENT                                                         001029  34100 11.11.14.063
401 C                   IF        COUNTER = 0                                 B01       001029  34200 11.11.14.063
                                        0
402 C                   Z-ADD     3             COUNTER                        01       001029  34300 11.11.14.063
                                                      3
403 C                   ENDIF                                                 E01       001029  34400 11.11.14.063
404  *----------------------------------------------------------------------            001029  34500 11.11.14.063
405 C                   if        counter > 0                                 B01       010522  34600 11.11.14.063
                                        3
406 C                   endif                                                 E01       010522  34700 11.11.14.063
407  *----------------------------------------------------------------------            010522  34800 11.11.14.063
408 C                   if        counter <> 0                                B01       010522  34900 11.11.14.064
                                        3
409 C                   endif                                                 E01       010522  35000 11.11.14.064
410  *----------------------------------------------------------------------            010522  35100 11.11.14.064
411  * AUDIT RPGIV EXTENDED FACTOR 2 CONDITIONAL AND/OR COMPLEX STATEMENTS              010602  35200 11.11.14.064
412 C                   Z-ADD     2             COUNTER                                 010602  35300 11.11.14.064
                                                      2
413 C                   Z-ADD     7             answer                                  010602  35400 11.11.14.064
                                                     7
414 C                   Z-ADD     *zero         final             7 2                   010607  35500 11.11.14.064
                                                  .00
415 C                   Z-sub     *zero         sum               6 1                   010704  35600 11.11.14.064
                                                 .0
416 C                   Z-ADD     3115          total             8 0                   010604  35700 11.11.14.064
                                                 3115
417 C                   Z-ADD     112           net               3 0                   010607  35800 11.11.14.067
                                                112
418 C                   IF        COUNTER = 0                                 B01       010602  35900 11.11.14.067
                                        2
419 C                             OR COUNTER = 2                              B01       010602  36000 11.11.14.067
                                           2
420 C                             OR COUNTER = 4                              B01       010602  36100 11.11.14.067
                                           2
421 C                             AND ANSWER = 7                              B01       010602  36200 11.11.14.067
                                           7
446 C                                     CCCC - DIFFERENCE +160 + EXTRA +              010617  38700 11.11.14.067
                                          7.12      1444.20           87
447 C                                     188  - INTERIM + EXTRA2                       010617  38800 11.11.14.067
                                                 1229.85      105
448 C                                   - 33.15 + GROSS                                 010617  38900 11.11.14.067
                                                 163.23
449 C                                   + MORE                                          010617  39000 11.11.14.067
                                         17.00
452 C                             + 44 - GROSS - EXTRA2                                 010617  39300 11.11.14.067
                                        163.23      105
422 C                   Z-ADD     3             COUNTER                        01       010604  36300 11.11.14.068
                                                      3
423 C                   ENDIF                                                 E01       010602  36400 11.11.14.068
424 C     sum           add       total         final                                   010604  36500 11.11.14.068
            .0                     3115
                                              3115.00
425 C                   eval      final = answer + counter + 5                          010602  36600 11.11.14.068
                                  15.00        7         3
```

```
426 C                   eval      sum = 4 + 6 - 2 + 1555                                010602  36700 11.11.14.068
                               1563.0
427 C                   eval      final = answer * counter                              010610  36800 11.11.14.068
                                  21.00        7         3
428 C                   eval      final = answer* counter                               010610  36900 11.11.14.068
                                  21.00        7        3
429 C                   eval      final = answer *counter                               010610  37000 11.11.14.068
                                  21.00        7        3
430 C                   eval      final = answer*counter                                010610  37100 11.11.14.068
                                  21.00        7       3
431 C                   eval      final = answer / counter                              010610  37200 11.11.14.069
                                   2.33        7         3
432 C                   eval      final = answer/counter                                010610  37300 11.11.14.069
                                   2.33        7       3
433 C                   eval      sum = 4 + 6 - counter + 1555                          010605  37400 11.11.14.069
                               1562.0                 3
434 C                   eval        total = 4 + 6 + sum                                 010603  37500 11.11.14.069
                                     1572        1562.0
435 C                   eval                    sum = 4 + 6                             010603  37600 11.11.14.069
                                               10.0
436 C                   eval      total = 4 + sum + 6 + answer + final - net            010606  37700 11.11.14.069
                                     82-     10.0            7
                                                                  2.33   112
437  *---------------------------------------------------------------------             010602  37800 11.11.14.069
438 C                   Z-ADD     7.12          cccc              5 2                   010617  37900 11.11.14.069
                                                7.12
                                                                                                      11.10.48.733   PAGE
439 C                   Z-ADD     163.23        gross             6 2                   010617  38000 11.11.14.069
                                               163.23
440 C                   z-add     1444.2        difference        8 2                   010617  38100 11.11.14.069
                                                   1444.20
441 C                   z-add     43.8          interim           8 2                   010617  38200 11.11.14.069
                                                  43.80
442 C                   z-add     87            extra             5 0                   010617  38300 11.11.14.069
                                                   87
443 C                   z-add     105           extra2            5 0                   010617  38400 11.11.14.069
                                                   105
444 C                   z-add     17            more              8 2                   010617  38500 11.11.14.069
                                               17.00
445 C                   eval      final = answer + counter +                            010617  38600 11.11.14.069
                                 783.80-       7         3
402 C                                     CCCC - DIFFERENCE +160 + EXTRA +              010617  39000 11.11.14.069
                                          7.12      1444.20           87
403 C                                     188  - INTERIM + EXTRA2                       010617  39000 11.11.14.069
                                                   43.80      105
404 C                                   - 33.15 + GROSS                                 010617  39000 11.11.14.069
                                                 163.23
405 C                                   + MORE                                          010617  39000 11.11.14.069
                                         17.00
450 C                   eval      net  = counter + cccc                                 010617  39100 11.11.14.069
                                   10          3   7.12
451 C                   eval      interim = answer - cccc + net + difference            010617  39200 11.11.14.069
                                  1229.85        7   7.12    10      1444.20
408 C                             + 44 - GROSS - EXTRA2                                 010617  39300 11.11.14.069
                                        163.23      105
453  *---------------------------------------------------------------------             010617  39400 11.11.14.069
454 C                   movel     '2'           movsw             1                     010604  39500 11.11.14.069
                                                2
455  * AUDIT RPGIV IF STATEMENT with alpha extended factor 2                            010604  39600 11.11.14.069
456 C                   if        movsw = '2'                               B01         010604  39700 11.11.14.069
                                  2
457 C                             OR MOVSW = '3'                            B01         010604  39800 11.11.14.069
                                     2
458 C                   movel     '1'           movsw             1          01         010604  39900 11.11.14.069
                                                1
459 C                   endif                                               E01         010604  40000 11.11.14.069
460 C                   movel     '5'           movsw             1                     010604  40100 11.11.14.069
                                                5
461 C                   if        movsw = '5'                               B01         010604  40200 11.11.14.069
                                  5
462 C                   endif                                               E01         010604  40300 11.11.14.069
463  *---------------------------------------------------------------------             010604  40400 11.11.14.069
```

```
464 C                   Z-ADD     2             COUNTER                                 010615  40500 11.11.14.069
                                                      2
465 C                   Z-ADD     7             answer                                  010615  40600 11.11.14.069
                                                     7
466 C                   Z-ADD     14.2          gggggg           6 2                    010615  40700 11.11.14.069
                                                 14.20
467 C                   IF        COUNTER = 0 and  gggggg =5 and              B01       010615  40800 11.11.14.069
                                        2           14.20
468 C                                COUNTER > 5 AND                          B01       010615  40900 11.11.14.069
                                           2
469 C                                COUNTER < 3 OR ANSWER = 6 AND            B01       010615  41000 11.11.14.069
                                           2             7
470 C                                COUNTER < 2                              B01       010615  41100 11.11.14.069
                                           2
471 C                             OR COUNTER = 2                              B01       010615  41200 11.11.14.069
                                           2
472 C                             AND ANSWER = 7 OR                           B01       010615  41300 11.11.14.069
                                           7
473 C                             ANSWER = 5                                  B01       010615  41400 11.11.14.069
                                       7
474 C                   Z-ADD     3             COUNTER                        01       010615  41500 11.11.14.070
                                                      3
475 C                   ENDIF                                                 E01       010615  41600 11.11.14.070
476  *                                                                -----             010615  41700 11.11.14.070
477  *---------------------------------------------------------------------             010615  41800 11.11.14.070
478  * AUDIT RPGIV MOVEL PADDED WITH BLANKS STATEMENT                                   010522  41900 11.11.14.070
479 C                   MOVEL     *ALL'M'       TESTML           20                     001029  42000 11.11.14.070
                                                MMMMMMMMMMMMMMMMMMMM
480 C                   MOVEL(P)  'LEFT'        TESTML                                  001029  42100 11.11.14.070
                                                LEFT
481  *---------------------------------------------------------------------             010330  42200 11.11.14.070
482  * all lower case statements                                                        010330  42300 11.11.14.070
483 C                   movel     *ALL'L'       testlo           20                     010330  42400 11.11.14.070
                                                LLLLLLLLLLLLLLLLLLLL
                                                                                                      11.10.48.733   PAGE
484 C                   z-add     11111         aaaaaaaaaa       6 0                    010330  42500 11.11.14.070
                                                     11111
485 C                   z-add     2222222       bbbbbbbbbb       8 3                    010618  42600 11.11.14.070
                                                 22222.000
486 C     aaaaaaaaaa    add       bbbbbbbbbb    cccccccccc       8 0                    010330  42700 11.11.14.070
               11111               22222.000
                                                     33333
487  *---------------------------------------------------------------------             001029  42800 11.11.14.070
488  * AUDIT RPGIV TIME STATEMENT                                                       001029  42900 11.11.14.070
489 C                   TIME                    TIMENOW          6 0                    001029  43000 11.11.14.070
                                                 111114
490  *---------------------------------------------------------------------             001029  43100 11.11.14.070
491 C                   movel     '2'           movsw            1                      010530  43200 11.11.14.070
                                                2
492  * AUDIT RPGIV IF STATEMENT                                                         010113  43300 11.11.14.070
493 C                   if        movsw = *blanks                             B01       010113  43400 11.11.14.070
                                  2
494 C                             OR COUNTER = 0                              B01       010113  43500 11.11.14.070
                                           3
496 C                   endif                                                 E01       010113  43700 11.11.14.070
497  *---------------------------------------------------------------------             010113  43800 11.11.14.070
499 C     100           DIV       5.25          NET              3 0           01       000323  48400 11.11.14.070
                                                 19
500 C                   MVR                     FRACT            4 4           01       000323  48500 11.11.14.070
                                                .2500
502  *  GOT ORDER DETAIL, PRINT IT                                                      000514  44300 11.11.14.070
503 C                   EXCEPT    PRTDET                              234               000514  44400 11.11.14.070
505  * GET THE CUSTOMER MASTER STORE RECORD                                             991225  44600 11.11.14.070
506 C                   Z-ADD     ODCUST        CUCUST                                  991225  44700 11.11.14.070
                                    1000
                                                  1000
507 C                   Z-ADD     ODSTOR        CUSTOR                                  991225  44800 11.11.14.070
                                       1
                                                     1
508  *----------------------------------------------------------------                  010118  44900 11.11.14.070
509  * THIS ROUTINE CAUSES THE WRONG CUSTOMER TO BE DISPLAYED                           010118  45000 11.11.14.070
510  * IF THE ORDER DETAIL FILE HAS BEEN UPDATED TWICE, AND ONLY TWICE                  010118  45100 11.11.14.070
```

```
 511  * FIND THE ERROR ON THE AUDIT REPORT BY SCANNING FOR 2050                      010118  45200 11.11.14.070
 512 C     UPDREC        IFEQ      2                                         B01       010118  45300 11.11.14.070
          2
 513 C                   ADD       1050          CUCUST                       01       051021  45400 11.11.14.070
                                                   2050
 514 C                   Z-ADD     1             CUSTOR                       01       010118  45500 11.11.14.070
                                                      1
 515 C                   MOVEL     *ALL'A'       @MSGDA                       01       051007  45600 11.11.14.070

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 516 C                   MOVEL     *ALL'B'       @MSGDB                       01       051007  45700 11.11.14.070

BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
 517  * call with parms                                                              040918  45800 11.11.14.070
 518 C                      CALL       'BATCHPGM1'                           GRAM    01        051007  45900 11.11.14.070
11.11.14.114
 519 C                   PARM                    @MSGDA           79          01       040624  46000 11.11.14.114

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 520 C                   PARM                    @MSGDB           79          01       040624  46100 11.11.14.114

BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
 521 C                   ENDIF                                               E01       010118  46200 11.11.14.114
 523  *----------------------------------------------------------------                                        010118  46400
11.11.14.114
 524 C     CUSKEY        CHAIN     CUSTREC1                          30        30 IS NOT000717  46500 11.11.14.114
       N30 00020500000001
CUCUST-0002050  CUSTOR-0000001  CUNAME-XYZ  STORE  -  ARDMORE          CUAD1-122  MONTGOMERY  AVE            CUAD2-THIRD  FLOOR
CUSTA-PA
 525 C                   z-add     *all'1'       aa                3 0                 050102  46600 11.11.14.114
                                                111
 526 C                   z-add     *all'2'       bb                3 0                 050102  46700 11.11.14.114
                                                222
 527 C                   z-add     *all'3'       cc                3 0                 050102  46800 11.11.14.114
                                                333
 528 C                   z-add     *all'4'       dd                3 0                 050102  46900 11.11.14.114
                                                444
 529 C                   z-add     *all'5'       ee                3 0                 050102  47000 11.11.14.114
                                                555
 530 C                   z-add     *all'6'       ff                3 0                 050102  47100 11.11.14.114
                                                666
 531 C                   z-add     *all'7'       gg                3 0                 050102  47200 11.11.14.117
                                                777
 532 C                   z-add     *all'8'       hh                3 0                 050102  47300 11.11.14.117
                                                888
 533 C                   z-add     *all'9'       ii                3 0                 050102  47400 11.11.14.117
                                                999
 534 C                   z-add     *zeros        total             8 0                 050102  47500 11.11.14.117
                                                     0
                                                                                                       11.10.48.733   PAGE
 536   total = aa + bb + cc + dd + ee + ff + gg + hh + ii;                               050102  47700 11.11.14.117
        4995  111  222  333  444  555  666  777  888  999
 538 c                   eval      total = aa + bb + cc + dd + ee + ff + gg + hh         050102  47900 11.11.14.117
                                    3996  111  222  333  444  555  666  777  888
 539 C     *IN30         IFEQ      *OFF                                      B01       991225  48000 11.11.14.117
           0
 540  * GOT CUSTOMER MASTER                                                            991225  48100 11.11.14.117
 541 C                   Z-ADD     CUCUST        KCUSNO                       01       001007  48200 11.11.14.117
                                     2050
                                                   2050
 542 C                   Z-ADD     CUSTOR        KSTORE                       01       010118  48300 11.11.14.117
                                        1
                                                      1
 543 C                   MOVEL     CUNAME        KCUSNA                       01       000323  48400 11.11.14.117
                                   XYZ STORE - ARDMORE
                                                 XYZ STORE - ARDMORE
 544 C                   MOVEL     CUNAME        PCUSNA                       01       000323  48500 11.11.14.117
                                   XYZ STORE - ARDMORE
                                                 XYZ STORE - ARDMORE
 545 C                   EXCEPT    PRTCUS                                     01       000323  48600 11.11.14.117
 548 C                   ENDIF                                       OFF     E01       991225  48900 11.11.14.117
 550  * DISPLAY DETAIL SCREEN                                                          000323  49100 11.11.14.117
 551 C     DISP02        TAG                                                             000514  49200 11.11.14.117
 552 C                   TIME                    TIMEN             6 0          START TIM010501  49300 11.11.14.117
                                                111114
 553 C                   EXFMT     NEWEXPD2                                              051007  49400 11.11.14.117 WRITE
*IN03-0 *IN43-0 EXPMDY-011807 KCUSNO-0002050 KCUSNA-XYZ STORE - ARDMORE          UDATE-120906 KSTORE-0000001 TIMEN-111114
KORDER-0001500 KLINE-00002
 553 C                   EXFMT     NEWEXPD2                                              051007  49400 11.11.14.495 READ
*IN03-0 *IN43-0 EXPMDY-011807 KCUSNO-0002050 KCUSNA-XYZ STORE - ARDMORE          UDATE-120906 KSTORE-0000001 TIMEN-111114
KORDER-0001500 KLINE-00002
 554  * TEST F3                                                                          000323  49500 11.11.14.495
 555 C     *IN03         CABEQ     *ON           DONE                                    000323  49600 11.11.14.495
           0
 557  * VALIDATE CHANGED DATE, AND UPDATE ORDER DETAIL                                   000323  49800 11.11.14.495
 559  * CONVERT EXPMDY FORMAT MMDDYY TO ODEXPD FORMAT YYYYMMDDYY                         000323  50000 11.11.14.495
 561 C                   Z-ADD     EXPMDY        YY                2 0          YY       000323  50200 11.11.14.495
                                    11807
                                                  7
 562 C     EXPMDY        DIV       100           MMDD              4 0          MMDD     000323  54700 11.11.14.495
            11807                                 118
 563 C     YY            MULT      10000         Y4MMDD            8 0          00YY0000 000323  50400 11.11.14.495
            7                                     70000
 564 C                   ADD       MMDD          Y4MMDD                         00YYMMDD 000323  50500 11.11.14.495
                                    118
                                                  70118
 565 C     YY            IFGT      40                                          B01       000323  50600 11.11.14.495
            7
 567 C                   ELSE                                                  X01       000323  50800 11.11.14.495
 568 C                   ADD       20000000      Y4MMDD                         01       000323  50900 11.11.14.495
                                               20070118
 569 C                   END                                                   E01       000323  51000 11.11.14.495
 570 C                   Z-ADD     Y4MMDD        ODEXPD                                  000323  51100 11.11.14.495
                                 20070118
                                               20070118
 571  * COMPLEX IF STATEMENT                                                             010614  51200 11.11.14.495
 572 C                   IF        COUNTER = 0 and                                     B01       010614  51300 11.11.14.495
                                         3
 573 C                                COUNTER > 5 AND                                  B01       010614  51400 11.11.14.495
                                            3
 574 C                                COUNTER < 3 OR ANSWER = 6 AND                    B01       010614  51500 11.11.14.495
                                            3             7
 575 C                                COUNTER < 2                                      B01       010614  51600 11.11.14.495
                                            3
 576 C                             OR COUNTER = 2                                      B01       010614  51700 11.11.14.495
                                            3
 577 C                             AND ANSWER = 7 OR                                   B01       010614  51800 11.11.14.495
                                            7
 578 C                             ANSWER = 5                                          B01       010614  51900 11.11.14.495
                                        7
 580 C                   ENDIF                                                 E01       010614  52100 11.11.14.495
 582  * TEST FOR FIELD OVER 100...                                                       010411  52300 11.11.14.495
 583 C                   MOVE      *ALL'#'       ALL#            256                     010411  52400 11.11.14.495
VAR     ALL#                                                                                      1     -                100
####################################################################################################
VAR      ALL#                                                                                 101      -                 200
####################################################################################################
VAR ALL#              201 -   256  ########################################################
 584 C                   MOVE      ALL#          ALL$            256                     010411  52500 11.11.14.495
VAR     ALL#                                                                                      1     -                100
####################################################################################################
VAR      ALL#                                                                                 101      -                 200
####################################################################################################
VAR ALL#              201 -   256  ########################################################
VAR     ALL$                                                                                      1     -                100
####################################################################################################
VAR      ALL$                                                                                 101      -                 200
####################################################################################################
VAR ALL$              201 -   256  ########################################################
                                                                                                       11.10.48.733     PAGE
23
 586 C                   MOVE      '2'           MOVSW                                   010429  52700 11.11.14.495
                                                 2
 587 C                   MOVE      '3'           MOVSW1            1                     010429  52800 11.11.14.495
```

```
                                                 3
 588 C                   MOVE      '4'           MOVSW2            1                     010429  52900 11.11.14.495
                                                 4
 589 C                   MOVE      '5'           MOVSW3            1                     010429  53000 11.11.14.495
                                                 5
 590 C                   MOVE      '5'           MOVSW4            1                     010429  53100 11.11.14.495
                                                 5
 591 C                   MOVE      'A'           HLD1              1                     010429  53200 11.11.14.495
                                                 A
 593 C     EXPMDY        IFEQ      UDATE                                       B01       010420  53400 11.11.14.495
            11807                  120906
 594 C     MOVSW         ANDEQ     *BLANK                                       01       000514  53500 11.11.14.495
           2
 595 C     MOVSW         OREQ      '5'                                          01       010429  53600 11.11.14.495
           2
 596 C     MOVSW1        OREQ      MOVSW2                                       01       010429  53700 11.11.14.495
           3                       4
 597 C     MOVSW3        ORNE      MOVSW4                                       01       010429  53800 11.11.14.495
           5                       5
 598 C     MOVSW         OREQ      '8'                                          01       010429  53900 11.11.14.495
           2
 599 C     MOVSW         OREQ      '9'                                          01       010429  54000 11.11.14.495
           2
 600 C     MOVSW         OREQ      'C'                                          01       010429  54100 11.11.14.495
           2
 601 C     MOVSW         OREQ      'D'                                          01       010429  54200 11.11.14.495
           2
 602 C     MOVSW         OREQ      HLD1                                         01       010429  54300 11.11.14.495
           2                       A
 603 C     MOVSW3        ANDNE     MOVSW2                                       01       010429  54400 11.11.14.495
           5                       4
 605 C                   END                                               E01       000323  54600 11.11.14.496
 607  * UPDATE ORDER DETAIL EXPECTED SHIP DATE                                           000323  54800 11.11.14.496
 608 C                   ADD       1             UPDREC            6 0          CNT ORD D010118  54900 11.11.14.496
                                                      3
 609 C                   UPDATE    ODETREC                             UND               000323  55000 11.11.14.496
ODORD#-0001500 ODLINE-00002 ODCUST-0001000 ODSTOR-0000001 ODITEM-Y2430          ODPRIC-0002515 ODQTY-0000003 ODREQD-20000317
ODEXPD-20070118 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
 611  * WRITE UPDATED ORDER DETAIL TO A WORK FILE                                        000402  55200 11.11.14.496
 612 C                   CLEAR                   ODETWRK                                 000402  55300 11.11.14.496
 613 C                   Z-ADD     ODORD#        WDORD#                                  000402  55400 11.11.14.496
                                     1500
                                                   1500
 614 C                   Z-ADD     ODLINE        WDLINE                                  000402  55500 11.11.14.496
                                        2
                                                      2
 615 C                   Z-ADD     ODCUST        WDCUST                                  000402  55600 11.11.14.496
                                     1000
                                                   1000
 616 C                   Z-ADD     ODSTOR        WDSTOR                                  000402  55700 11.11.14.496
                                        1
                                                      1
 617 C                   MOVEL     ODITEM        WDITEM                                  000402  55800 11.11.14.496
                                   Y2430
                                                 Y2430
 618 C                   Z-ADD     ODPRIC        WDPRIC                                  000402  55900 11.11.14.496
                                    25.15
                                                  25.15
 619 C                   Z-ADD     ODQTY         WDQTY                                   000402  56000 11.11.14.496
                                       3
                                                     3
 620 C                   Z-ADD     ODREQD        WDREQD                                  000402  56100 11.11.14.496
                                 20000317
                                               20000317
 621 C                   Z-ADD     ODEXPD        WDEXPD                                  000402  56200 11.11.14.496
                                 20070118
                                               20070118
 622 C                   Z-ADD     ODSHPD        WDSHPD                                  000402  56300 11.11.14.496
                                        0
                                                      0
 623 C                   Z-ADD     ODINV#        WDINV#                                  000402  56400 11.11.14.496
                                        0
```

```
                                                      0
 624 C                   MOVEL     ODSTAT        WDSTAT                                  000402  56500 11.11.14.496
                                   O
                                                 O
                                                                                                       11.10.48.733   PAGE
 625 C                   MOVEL     ODX           WDX                                     000402  56600 11.11.14.496
 627  * WRITE ORDERWK                                                                    000402  56800 11.11.14.496
 628 C                   WRITE     ODETWRK                                               000402  56900 11.11.14.496
WDORD#-0001500 WDLINE-00002 WDCUST-0001000 WDSTOR-0000001 WDITEM-Y2430                  WDPRIC-0002515 WDQTY-0000003 WDREQD-
20000317 WDEXPD-20070118 WDSHPD-00000000 WDINV#-0000000 WDSTAT-O WDX-
 630  * NESTED IF STATEMENTS WITH AND/OR                                                 010429  57100 11.11.14.496
 631 C     aaaaaaaaaa    ifeq      bbbbbbbbbb                              B01           010429  57200 11.11.14.496
                11111               22222.000
 632 C     UPDREC        andeq     YY                                       01           010429  57300 11.11.14.496
                3                   7
 637 C                   END                                         Q     E01           010429  57800 11.11.14.496
 639  * REDISPLAY FIRST SCREEN                                                           000323  58000 11.11.14.496
 640 C                   GOTO      DISP01                                                000514  58100 11.11.14.496
 287 C     DISP01        TAG                                                             000514  22800 11.11.14.496
 288  * CLEAR EXPECTED SHIP DATE AND ERROR CODE                                          000514  22900 11.11.14.496
 289 C                   Z-ADD     *ZERO         PEXPSH                                  000514  23000 11.11.14.496
                                                      0
 290 C                   MOVEL     *BLANKS       PERROR                                  000514  23100 11.11.14.496
 291 C                   Z-ADD     *ZEROS        KCUSNO                                  001002  23200 11.11.14.496
                                                      0
 292 C                   Z-ADD     *ZEROS        KSTORE                                  001002  23300 11.11.14.496
                                                      0
 293 C                   MOVEL     *BLANKS       KCUSNA                                  000323  23400 11.11.14.496
 294 C                   Z-ADD     *ZERO         EXPMDY                                  000323  23500 11.11.14.497
                                                      0
 295 C                   TIME                    TIMEN            6 0          START TIM010501  23600 11.11.14.498
                                                111114
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.11.14.498 WRITE
*IN03-0 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111114
 296 C                   EXFMT     NEWEXPD1                                              051007  23700 11.11.32.474 READ
*IN03-1 *IN42-0 KORDER-0001500 KLINE-00002 UDATE-120906 TIMEN-111114
 297  * TEST F3                                                                          000323  23800 11.11.32.474
 298 C     *IN03         CABEQ     *ON           DONE                                    000323  23900 11.11.32.474
           1
 645 C     DONE          TAG                                                             000323  58600 11.11.32.474
 646 C                   movel     'xxxxxxxxxxxx'alphxxxxxxxxxx   12                     010708  58700 11.11.32.475
                                                 xxxxxxxxxxxx
 647 C                   movel     'yyyyyyyyyyyy'alphyyyyyyyyyy   12                     010708  58800 11.11.32.475
                                                 yyyyyyyyyyyy
 648 C                   movel     'ssssssssssss'alphssssssssss   12                     010708  58900 11.11.32.475
                                                 ssssssssssss
 649 C                   movel     'tttttttttttt'alphtttttttttt   12                     010708  59000 11.11.32.475
                                                 tttttttttttt
 651 C                   IF        alphxxxxxxxxxx = alphyyyyyyyyyy or          B01       010708  59200 11.11.32.475
                                   xxxxxxxxxxxx     yyyyyyyyyyyy
 652 C                                ALPHSSSSSSSSSS = ALPHTTTTTTTTTT          B01       010708  59300 11.11.32.475
                                      ssssssssssss     tttttttttttt
 653 C                   end                                               E01       010708  59400 11.11.32.475
 655 c                   movel     message(1)    text1                                   020623  59600 11.11.32.475
                                                 Invalid Account Number                           0
VAR MESSAGE(1)            1 -     50  Invalid Account Number                            0See Main Store Account Number
0
 656 c                   eval      text2 = message(2)                                    020623  59700 11.11.32.475
                                   See Main Store Account Number                 0
                                           See Main Store Account Number                0
 657 c                   eval      text5 = message(10)                                    20623  59800 11.11.32.476
                                   Sales HOLD - Not authorized to access         1
                                           Sales HOLD - Not authorized to access        1
 659  * CALL A BOUND MODULE                                                          PH*** 010127  60000 11.11.32.476
 660  ****************   CALLB     'PROCXYZ'                                         PH*** 040624  60100 11.11.32.476
 663 C                   MOVE      *ALL'A'       @MSGDA                                  040624  60400 11.11.32.476

AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
 664 C                   MOVE      *ALL'B'       @MSGDB                                  040624  60500 11.11.32.476

BBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBBB
 666 C                   Z-ADD     514.22        GROSSAAAAAAAAA    7 2                   010807  60700 11.11.32.476
```

```
                                                         514.22
 667 C                   Z-ADD     40            HOURSBBBBBBBBB    6 3                   010807  60800 11.11.32.476
                                                         40.000
 668 C                   Z-ADD     12345678      ODEXPD            8 0                   010807  60900 11.11.32.476
                                               12345678
 670 C     ODEXPD        DIV       10000         EXPYY             2 0                   010703  65500 11.11.32.476
         12345678                                   34
 672 C     100           DIV       5.25          NET               3 0                   010703  65700 11.11.32.476
                                                  19
 673 C                   MVR                     FRACT             4 4                   010703  65800 11.11.32.476
                                                 .2500
                                                                                                       11.10.48.733   PAGE
 675 C     GROSSAAAAAAAAAADIV      HOURSBBBBBBBBBRATECCCCCCCCCC    5 2          01       010624  66000 11.11.32.477
                   514.22                  40.000         12.85
 676 C                   MVR                     LEFTDDDDDDDDDD    4 4          01       010624  66100 11.11.32.477
                                                          .2200
 678 C     ODEXPD        DIV       10            YYYY              4 0          01       010624  66300 11.11.32.477
         12345678                                4567
 679  *----------------------------------------------------------------                  010113  62000 11.11.32.477
 680  * READ AND PRINT ALL CUSTOMER STORE RECORDS FOR DISPLAYED CUST.                    010113  62100 11.11.32.477
 681  *----------------------------------------------------------------                  010113  62200 11.11.32.477
 683 C     CUCUST        SETLL     CUSTREC1                                              010113  62400 11.11.32.477
           0002050
 684 C     CUCUST        READE     CUSTREC1                               84             010113  62500 11.11.32.477
       N84 0002050
CUCUST-0002050  CUSTOR-0000000  CUNAME-XYZ  CORPORATE  OFFICE            CUAD1-555  ARCH  STREET                      CUAD2-
CUSTA-PA
 685 C     *IN84         DOWEQ     *OFF                                        B01       010113  62600 11.11.32.477
           0
 686 C                   EXCEPT    PRTCUS                                       01       010113  62700 11.11.32.477
 687  * READ ANOTHER RECORD                                                              010113  62800 11.11.32.477
 688 C     CUCUST        READE     CUSTREC1                               84    01       010113  62900 11.11.32.477
       N84 0002050
CUCUST-0002050  CUSTOR-0000001  CUNAME-XYZ  STORE  -  ARDMORE          CUAD1-122  MONTGOMERY  AVE            CUAD2-THIRD  FLOOR
CUSTA-PA
 685 C     *IN84         DOWEQ     *OFF                                        B01       010113  62600 11.11.32.477
           0
 686 C                   EXCEPT    PRTCUS                                       01       010113  62700 11.11.32.477
 687  * READ ANOTHER RECORD                                                              010113  62800 11.11.32.477
 688 C     CUCUST        READE     CUSTREC1                               84    01       010113  62900 11.11.32.477
        84 0002050
 689 C                   ENDDO                                            OF E01       010113  63000 11.11.32.477
 691 C                      CALL       'Z$PGM01C'                                      GET USER 000402  63200 11.11.32.477
11.11.32.477
 692 C                   PARM                    USERNA           50                     000402  63300 11.11.32.477
                                                 RTPA - paul harkins
 693 C                   MOVEL     USERNA        KPGMRN           50            PROGRAMM 000402  63400 11.11.32.478
                                   RTPA - paul harkins
                                                 RTPA - paul harkins
 695  * EXIT PROGRAM                                                                     000402  63600 11.11.32.478
 696 C                   Z-ADD     EXPMDY        PEXPSH                         EXP SHIP 000323  63700 11.11.32.478
                                        0
                                                      0
 697 C                   MOVEL     PARMRE        PARMIN                         NOW 4 FIE000302  63800 11.11.32.478
                                   000150000001000000XYZ STORE - ARDMORE
                                                 000150000001000000XYZ STORE - ARDMORE
 700 C                   Z-ADD     1             i                 3 0                   010703  64100 11.11.32.478
                                                 1
 701 C                   Z-ADD     3             J                 3 0                   010703  64200 11.11.32.478
                                                 3
 702 C                   Z-ADD     7             kkkkkkkkk         3 0                   010703  64300 11.11.32.478
                                                         7
 703 C                   Z-ADD     13            lllllllll         3 0                   010703  64400 11.11.32.478
                                                        13
 704 C                   Z-ADD     05            iiiiiiiii         3 0                   010703  64500 11.11.32.479
                                                         5
 705 C                   Z-ADD     123           $$D(05)                                 010703  64600 11.11.32.479
                                                      23
 706 C                   Z-ADD     6             $$D(4)                                  010703  64700 11.11.32.479
                                                      6
 707 C                   Z-ADD     18            $$D(3)                                  010703  64800 11.11.32.479
                                                     18
 709 C                   eval                    final = answer + counter                010703  65000 11.11.32.479
                                                 10.00        7         3
 711 C                   eval      $$d(i) = answer + counter                             010703  65200 11.11.32.479
                                       10        7         3
 713 C                   eval      final = answer + counter +6                           010703  65400 11.11.32.479
                                   16.00        7         3
 714 C                   eval      $$d(kkkkkkkkk) = $$D(I) + $$D(5) +COUNTER             010703  65500 11.11.32.479
                                               36       10       23        3
 715 C                   eval        $$d(2) = $$D(I) + counter +$$D(J)                   010703  65600 11.11.32.479
                                         31       10         3      18
 716 C                   eval        $$d(2) = $$D(I) + $$D(4) + $$D(J)                   010703  65700 11.11.32.479
                                         34       10        6       18
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.479
                                                 1
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.479
                                   1
                                                      1
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.481
                                                     33
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.481
            33                                        56
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.481
            33                          56
                                                      89
                                                                                                       11.10.48.733   PAGE
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.481
           33341806231236121212121212
                                   056000000000000000000000000000000000000

18480000000000000000000000000000000000000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.481
                                       33
                                                  66.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.481
            33                                  165.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.481
           1
 730 C                   END                                                   E02       010624  67100 11.11.32.481
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.481
            33
 734 C                   END                                                   E02       010624  67500 11.11.32.482
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.482
            33                          56
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.482
            34                           0
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.482
             56                         89
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.482
           1848                          0
 741 C                   END                                                   E02       010624  68200 11.11.32.483
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.483
            33
 745 C                   END                                                   E02       010624  68600 11.11.32.483
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.483
                                                 2
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.483
                                   2
                                                      2
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.483
                                                     34
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.483
            34                                        57
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.483
            34                          57
                                                      91
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.483
           33341806231236121212121212
                                   056057000000000000000000000000000000000

18481938000000000000000000000000000000000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.483
                                       34
```

```
                                                  68.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.483
            34                                  170.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.483
           2
 730 C                   END                                                   E02       010624  67100 11.11.32.483
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.483
            34
 734 C                   END                                                   E02       010624  67500 11.11.32.483
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.483
            34                          57
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.483
            34                           0
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.483
             57                         91
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.483
           1938                          0
 741 C                   END                                                   E02       010624  68200 11.11.32.485
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.485
            34
 745 C                   END                                                   E02       010624  68600 11.11.32.485
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.485
                                                 3
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.485
                                   3
                                                      3
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.485
                                                     35
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.485
            35                                        58
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.485
            35                          58
                                                      93
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.485
           333435062312361212121212l212
                                   056057058000000000000000000000000000000000

184819382030000000000000000000000000000000000000000000000000000000000000000000000
                                                                                                       11.10.48.733   PAGE
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.485
                                       35
                                                  70.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.485
            35                                  175.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.485
           3
 730 C                   END                                                   E02       010624  67100 11.11.32.485
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.485
            35
 734 C                   END                                                   E02       010624  67500 11.11.32.485
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.485
            35                          58
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.485
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.485
             58                         93
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.485
           2030                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.485
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.485
            35
 745 C                   END                                                   E02       010624  68600 11.11.32.485
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.485
                                                 4
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.485
                                   4
                                                      4
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.485
                                                     36
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.485
            36                                        59
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.486
```

```
            36                        59
                                                      95
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.486
           33343536231236121212121212
                                   056057058059000000000000000000000000000

18481938203021240000000000000000000000000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.486
                                       36
                                                  72.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.486
            36                                  180.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.486
           4
 730 C                   END                                                   E02       010624  67100 11.11.32.486
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.486
            36
 734 C                   END                                                   E02       010624  67500 11.11.32.486
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.486
            36                        59
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.486
            34                        58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.486
             59                       95
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.486
           2124                       93
 741 C                   END                                                   E02       010624  68200 11.11.32.486
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.486
            36
 745 C                   END                                                   E02       010624  68600 11.11.32.486
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.486
                                                 5
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.486
                                   5
                                                      5
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.486
                                                     37
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.486
            37                                        60
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.486
            37                        60
                                                      97
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.486
           33343536371236121212121212
                                   056057058059060000000000000000000000000

18481938203021242220000000000000000000000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.486
                                       37
                                                  74.0
                                                                                                       11.10.48.733   PAGE
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.486
            37                                  185.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.486
           5
 728 C                   MOVEL     'EEEEEEEE'    $$A(I)                         02       010624  66900 11.11.32.486
                                                 EEEEEEEE
 729 C                   MOVE      I             $$A(I)                         02       010624  67000 11.11.32.486
                                   5
                                                 EEEEEEE005
 730 C                   END                                                   E02       010624  67100 11.11.32.486
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.486
            37
 734 C                   END                                                   E02       010624  67500 11.11.32.486
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.486
            37                        60
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.486
            34                        58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.486
             60                       97
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.486
           2220                       93
 741 C                   END                                                   E02       010624  68200 11.11.32.486
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.486
            37
 745 C                   END                                                   E02       010624  68600 11.11.32.486
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.486
                                                 6
 720 C                   Z-ADD     I             $$D(I)                          01       010624  66100 11.11.32.486
                                   6
                                                      6
 721 C                   ADD       32            $$D(I)                          01       010624  66200 11.11.32.486
                                                     38
 722 C     $$D(I)        ADD       23            $$D2(I)                         01       010624  66300 11.11.32.486
            38                                        61
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                         01       010624  66400 11.11.32.486
            38                          61
                                                      99
 724 C     $$D           MULT      $$D2          $$D4                            01       010624  66500 11.11.32.487
           33343536373836121212121212
                                   056057058059060061000000000000000000000

18481938203021242220231800000000000000000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1           01       010624  66600 11.11.32.487
                                       38
                                                  76.0
 726 C     $$D(I)        MULT      5             WORK7             7 2           01       010624  66700 11.11.32.487
            38                                  190.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.487
           6
 730 C                   END                                                   E02       010624  67100 11.11.32.487
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.487
            38
 734 C                   END                                                   E02       010624  67500 11.11.32.487
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.487
            38                          61
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.487
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.487
             61                         99
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.487
           2318                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.487
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.487
            38
 745 C                   END                                                   E02       010624  68600 11.11.32.490
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.490
                                                 7
 720 C                   Z-ADD     I             $$D(I)                          01       010624  66100 11.11.32.490
                                   7
                                                      7
 721 C                   ADD       32            $$D(I)                          01       010624  66200 11.11.32.490
                                                     39
 722 C     $$D(I)        ADD       23            $$D2(I)                         01       010624  66300 11.11.32.490
            39                                        62
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                         01       010624  66400 11.11.32.490
            39                          62
                                                     101
 724 C     $$D           MULT      $$D2          $$D4                            01       010624  66500 11.11.32.490
           33343536373839121212121212
                                   056057058059060061062000000000000000000

18481938203021242220231824180000000000000000000000000000000000000000000000000000
                                                                                                       11.10.48.733   PAGE
 725 C     2             MULT      $$D(I)        WORK5             5 1           01       010624  66600 11.11.32.490
                                       39
                                                  78.0
 726 C     $$D(I)        MULT      5             WORK7             7 2           01       010624  66700 11.11.32.490
            39                                  195.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.490
           7
 730 C                   END                                                   E02       010624  67100 11.11.32.490
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.490
            39
 734 C                   END                                                   E02       010624  67500 11.11.32.490
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.490
            39                          62
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.490
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.490
             62                        101
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.490
           2418                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.492
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.492
            39
 745 C                   END                                                   E02       010624  68600 11.11.32.492
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.492
                                                 8
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.492
                                   8
                                                      8
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.492
                                                     40
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.492
            40                                        63
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.492
            40                          63
                                                     103
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.492
           33343536373839401212121212
                                   0560570580590600610620630000000000000000

18481938203021242220231824182520000000000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.492
                                       40
                                                  80.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.492
            40                                  200.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.492
           8
 730 C                   END                                                   E02       010624  67100 11.11.32.492
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.492
            40
 734 C                   END                                                   E02       010624  67500 11.11.32.492
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.492
            40                          63
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.492
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.492
             63                        103
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.492
           2520                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.492
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.492
            40
 745 C                   END                                                   E02       010624  68600 11.11.32.492
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.493
                                                 9
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.493
                                   9
                                                      9
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.493
                                                     41
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.493
            41                                        64
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.493
            41                          64
                                                     105
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.493
           33343536373839404112121212
                                   0560570580590600610620630640000000000000

18481938203021242220231824182520262400000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.493
                                       41
```

```
                                                  82.0
                                                                                                       11.10.48.733   PAGE
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.493
            41                                  205.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.493
           9
 730 C                   END                                                   E02       010624  67100 11.11.32.493
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.493
            41
 734 C                   END                                                   E02       010624  67500 11.11.32.493
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.493
            41                          64
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.493
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.493
             64                        105
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.493
           2624                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.493
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.493
            41
 745 C                   END                                                   E02       010624  68600 11.11.32.493
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.493
                                                10
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.493
                                  10
                                                     10
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.493
                                                     42
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.493
            42                                        65
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.493
            42                          65
                                                     107
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.493
           33343536373839404142121212
                                   0560570580590600610620630640650000000000

1848193820302124222023182418252026242730000000000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.493
                                       42
                                                  84.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.493
            42                                  210.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.493
          10
 730 C                   END                                                   E02       010624  67100 11.11.32.493
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.493
            42
 734 C                   END                                                   E02       010624  67500 11.11.32.493
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.493
            42                          65
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.493
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.493
             65                        107
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.493
           2730                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.493
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.493
            42
 745 C                   END                                                   E02       010624  68600 11.11.32.493
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.493
                                                11
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.493
                                  11
                                                     11
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.494
                                                     43
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.494
            43                                        66
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.494
```

```
            43                          66
                                                     109
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.494
           3334353637383940414243121 2
                                   056057058059060061062063064065066000000

1848193820302124222023182418252026242730283800000000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.494
                                       43
                                                  86.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.494
            43                                  215.00
                                                                                                       11.10.48.733   PAGE
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.494
          11
 730 C                   END                                                   E02       010624  67100 11.11.32.494
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.494
            43
 734 C                   END                                                   E02       010624  67500 11.11.32.494
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.494
            43                          66
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.494
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.494
             66                         109
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.494
           2838                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.494
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.494
            43
 745 C                   END                                                   E02       010624  68600 11.11.32.494
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.494
                                                12
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.494
                                  12
                                                     12
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.494
                                                     44
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.494
            44                                        67
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.494
            44                          67
                                                     111
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.494
           333435363738394041424344 12
                                   056057058059060061062063064065066067000

1848193820302124222023182418252026242730283829480000000000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.494
                                       44
                                                  88.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.494
            44                                  220.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.494
          12
 730 C                   END                                                   E02       010624  67100 11.11.32.494
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.494
            44
 734 C                   END                                                   E02       010624  67500 11.11.32.494
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.494
            44                          67
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.494
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.494
             67                         111
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.494
           2948                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.494
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.494
            44
 745 C                   END                                                   E02       010624  68600 11.11.32.501
 719 C                   DO        13            I                 3 0         B01       010624  66000 11.11.32.501
```

```
                                                13
 720 C                   Z-ADD     I             $$D(I)                         01       010624  66100 11.11.32.501
                                  13
                                                     13
 721 C                   ADD       32            $$D(I)                         01       010624  66200 11.11.32.501
                                                     45
 722 C     $$D(I)        ADD       23            $$D2(I)                        01       010624  66300 11.11.32.501
            45                                        68
 723 C     $$D(I)        ADD       $$D2(I)       $$D3(I)                        01       010624  66400 11.11.32.501
            45                          68
                                                     113
 724 C     $$D           MULT      $$D2          $$D4                           01       010624  66500 11.11.32.501
           33343536373839404142434445
                                   056057058059060061062063064065066067068

18481938203021242220231824182520262427302838294830600000000000000000000000000000
 725 C     2             MULT      $$D(I)        WORK5             5 1          01       010624  66600 11.11.32.501
                                       45
                                                  90.0
 726 C     $$D(I)        MULT      5             WORK7             7 2          01       010624  66700 11.11.32.501
            45                                  225.00
 727 C     I             IFEQ      5                                           B02       010624  66800 11.11.32.501
          13
                                                                                                       11.10.48.733   PAGE
 730 C                   END                                                   E02       010624  67100 11.11.32.501
 731 C     $$D(I)        IFEQ      7                                           B02       010624  67200 11.11.32.501
            45
 734 C                   END                                                   E02       010624  67500 11.11.32.501
 735 C     $$D(I)        IFEQ      $$D2(I)                                     B02       010624  67600 11.11.32.501
            45                          68
 736 C     $$D(2)        OREQ      $$D2(3)                                      02       010624  67700 11.11.32.501
            34                          58
 737 C     $$D2(I)       ANDNE     $$D3(I)                                      02       010624  67800 11.11.32.501
             68                        113
 738 C     $$D4(I)       ANDEQ     $$D3(3)                                      02       010624  67900 11.11.32.501
           3060                         93
 741 C                   END                                                   E02       010624  68200 11.11.32.501
 742 C     $$D(I)        IFEQ      9                                           B02       010624  68300 11.11.32.501
            45
 745 C                   END                                                   E02       010624  68600 11.11.32.501
 746 C                   ENDDO                                                 E01       010624  68700 11.11.32.501
 748 C                   exsr      moveit                                                010709  68900 11.11.32.501
 749 C                   exsr      move22                                                010709  69000 11.11.32.502
 937 C     MOVE22        BEGSR                                                           010709  87800 11.11.32.502
 938 C                   MOVEL     '22'          MOVSWx            2                     010709  87900 11.11.32.502
                                                 22
 939 C                   ENDSR                                                           010709  88000 11.11.32.503
 750 C                   exsr      move3333333333                                        010710  69100 11.11.32.503
 941 C     MOVE3333333333BEGSR                                                           010710  88200 11.11.32.503
 942 C                   MOVEL     '333333333333'MOVSWx33333333   12                     010710  88300 11.11.32.503
                                                 333333333333
 943 C                   ENDSR                                                           010710  88400 11.11.32.503
 751 C                   exsr      move4444444444                                        010710  69200 11.11.32.503
 945 C     MOVE4444444444BEGSR                                                           010710  88600 11.11.32.503
 946 C                   MOVEL     '444444444444'MOVSWx44444444   12                     010710  88700 11.11.32.503
                                                 444444444444
 947 C                   do        5             ll                3 0         B01 0917  020917  88800 11.11.32.503
                                                  1
 948 C                   add       1             qq                5 0          01 0917  020917  88900 11.11.32.503
                                                  1
 949 C     ll            ifeq      4                                           B02 0917  020917  89000 11.11.32.503
            1
 952 C                   endif                                                 E02 9917  020917  89300 11.11.32.503
 947 C                   do        5             ll                3 0         B01 0917  020917  88800 11.11.32.503
                                                  2
 948 C                   add       1             qq                5 0          01 0917  020917  88900 11.11.32.503
                                                  2
 949 C     ll            ifeq      4                                           B02 0917  020917  89000 11.11.32.503
            2
 952 C                   endif                                                 E02 9917  020917  89300 11.11.32.503
 947 C                   do        5             ll                3 0         B01 0917  020917  88800 11.11.32.503
                                                  3
 948 C                   add       1             qq                5 0          01 0917  020917  88900 11.11.32.503
                                                  3
 949 C     ll            ifeq      4                                           B02 0917  020917  89000 11.11.32.503
            3
 952 C                   endif                                                 E02 9917  020917  89300 11.11.32.503
 947 C                   do        5             ll                3 0         B01 0917  020917  88800 11.11.32.503
                                                  4
 948 C                   add       1             qq                5 0          01 0917  020917  88900 11.11.32.503
                                                  4
 949 C     ll            ifeq      4                                           B02 0917  020917  89000 11.11.32.504
            4
 950  *  leavesr leaves the subroutine                                             0917  020917  89100 11.11.32.504
 951 C                   leavesr                                                02 0917  020917  89200 11.11.32.504
 752 C                   MOVE      '1'           *IN25                                   010625  69300 11.11.32.504
                                                 1
 753 C                   Z-ADD     11111111111111aaaaaaaaaaaaaa   10 0                   010701  69400 11.11.32.504
                                                     1111111111
 754 C                   Z-ADD     22222222222222bbbbbbbbbbbbbb   13 2                   010701  69500 11.11.32.504
                                                 22222222222.00
 755 C                   Z-ADD     *zero         cccccccccccccc   10 1                   010701  69600 11.11.32.504
                                                             .0
 756 C     aaaaaaaaaaaaaaadd       bbbbbbbbbbbbbbcccccccccccccc                          010701  69700 11.11.32.504
               1111111111          22222222222.00
                                                    333333333.0
 757 C                   eval        bbbbbbbbbbbbbb = aaaaaaaaaaaaaa +122 + 14.2         010701  69800 11.11.32.505
                                      1111111247.20       1111111111
 758 C     *IN25         IFEQ      *ON                                         B01       010625  69900 11.11.32.505
           1
 759 C     *IN(27)       OREQ      *OFF                                         01       010625  70000 11.11.32.505
           0
 760 C     '1'           OREQ      *IN(27)                                      01       010625  70100 11.11.32.505

0
                                                                                                       11.10.48.733   PAGE
 761 C                   MOVE      '1'           *IN26                          01       010625  70200 11.11.32.505
                                                 1
 762 C                   END                                                   E01       010625  70300 11.11.32.505
 763 C     *IN25         IFEQ      *IN27                                       B01       010625  70400 11.11.32.505
           1                       0
 765 C                   END                                                   E01       010625  70600 11.11.32.505
 766 C                   MOVE      '1'           *IN25                                   010625  70700 11.11.32.505
                                                 1
 767 C                   MOVE      '1'           *IN(27)                                 010625  70800 11.11.32.505
                                                 1
 769 C                   Z-ADD     2             longindex         3 0                   010627  71000 11.11.32.505
                                                         2
 770 C                   Z-ADD     4             shorter           3 0                   010627  71100 11.11.32.505
                                                       4
 771 C     $$D(longindex)MULT      3             WORK5             5 1                   010627  71200 11.11.32.506
            34                                   102.0
 772 C     $$D(longindex)ifeq      2                                           B01       010627  71300 11.11.32.506
            34
 774 C                   end                                                   E01       010627  71500 11.11.32.506
 775 C     $$d(shorter)  add       $$d(longindex)$$D4(shorter)                           010627  71600 11.11.32.506
            36                                 34
                                                            70
 776  * test built-in functions BIFs                                                     010812  71700 11.11.32.506
 777  *---------------------------------------------------------------------             010812  71800 11.11.32.506
 778  *  %SUBST BIF                                                                      010812  71900 11.11.32.506
 779 C                   eval      Phone = '6105551212'                                  010812  72000 11.11.32.506
                                   6105551212
 781 C                   eval      AreaCode = %SUBST(Phone:1:3)                     %SUBST BI010812  72200 11.11.32.506
                                   610               6105551212
 782 C                   eval      Exchange = %SUBST(Phone:4:3)                     %SUBST BI010812  72300 11.11.32.506
                                   555               6105551212
 783 C                   eval      Local = %SUBST(Phone:7)                          %SUBST BI010812  72400 11.11.32.506
                                   1212           6105551212
 784  *---------------------------------------------------------------------             010812  72500 11.11.32.506
 785  *  %SCAN %REPLACE BIF                                                              010814  72600 11.11.32.506
 786 C                   eval      CusNm = 'Tom & Jerry'                            %SUBST B 010814  72700 11.11.32.507
                                   Tom & Jerry
 787 C                   eval      ScanX = %scan(' & ': CusNm)                      %SUBST B 010814  72800 11.11.32.507
```

```
                                       4                Tom & Jerry
 788 C                   if        ScanX > *zero                               B01       010814  72900 11.11.32.507
                                       4
 790 C                               %REPLACE(' AND ': CUSNM: SCANX: 3)         01       010814  73100 11.11.32.507
                                                       Tom & Jerry
                                                                  4
 789 C                   eval      CustNamev =                                  01       010814  73000 11.11.32.507
                                   Tom and Jerry
 746 C                               %REPLACE(' AND ': CUSNM: SCANX: 3)         01       010814  73100 11.11.32.507
                                                       Tom & Jerry
                                                                  4
 791 C                   endif                                                 E01       010814  73200 11.11.32.507
 792  *----------------------------------------------------------------------            010814  73300 11.11.32.507
 793  *  %LEN BIF                                                                        010814  73400 11.11.32.507
 794 C                   eval      %len(CustNamev) = 0                          %SUBST BI010814  73500 11.11.32.507
 795  *----------------------------------------------------------------------            010814  73600 11.11.32.507
 797  *  %SIZE BIF                                                                       010812  73800 11.11.32.507
 798  * obtain the size of a field                                                       010812  73900 11.11.32.507
 799 C                   eval      siz = %SIZE(fielda)                          %SIZE BI 010812  74000 11.11.32.507
                                    25
 800  * obtain the size of one array element                                             010812  74100 11.11.32.508
 801 C                   eval      siz = %SIZE(arrayx)                                   010812  74200 11.11.32.508
                                     2
                                               000000000000000000000000000000000000
 802  * obtain the size of an entire array                                               010812  74300 11.11.32.508
 803 C                   eval      siz = %SIZE(arrayx:*ALL)                              010812  74400 11.11.32.508
                                    36
                                               000000000000000000000000000000000000
 804  * obtain the size of a named literal                                               010812  74500 11.11.32.508
 805 C                   eval      siz = %SIZE(@V)                                       010812  74600 11.11.32.508
                                    38
 806  * obtain the size of a literal                                                     010812  74700 11.11.32.508
 807 C                   eval      siz = %SIZE('abcdef')                                 010812  74800 11.11.32.508
                                     6
 808  *----------------------------------------------------------------------            010812  74900 11.11.32.508
 809  *  %TRIML %TRIMR %TRIM BIF                                                         010812  75000 11.11.32.508
 810 C                   eval      Textline1 = '    abcdehghijklm    '                   010812  75100 11.11.32.508
                                       abcdehghijklm
 811 C                   eval      Textline2 = %TRIML(Textline1)                %TRIML   010812  75200 11.11.32.509
                                   abcdehghijklm
                                                          abcdehghijklm
                                                                                                       11.10.48.733   PAGE
 812 C                   eval      Textline2 = %TRIMR(Textline1)                %TRIMR   010812  75300 11.11.32.509
                                       abcdehghijklm
                                                          abcdehghijklm
 813 C                   eval      Textline2 = %TRIM(Textline1)                 %TRIM    010812  75400 11.11.32.509
                                   abcdehghijklm
                                                         abcdehghijklm
 815  *----------------------------------------------------------------------            010812  75600 11.11.32.509
 816  *  %OPEN BIF                                                                       010812  75700 11.11.32.509
 817 C                   if        not %open(qprint)                           B01       010812  75800 11.11.32.509
 818 C                   open (e)  qprint                                       01       010812  75900 11.11.32.512
 819 C                   if        %error                                      B02       010812  76000 11.11.32.512
 822 C                   else                                                  X02       010812  76300 11.11.32.512
 823  * good user controlled open                                                        010812  76400 11.11.32.512
 824 C                   TIME                    TIMES             6 0          02       010812  76500 11.11.32.512
                                                111132
 825 C                   EXCEPT    AUDSTRHEAD                                   02       010812  76600 11.11.32.512
 826 C                   end                                                   E02       010812  76700 11.11.32.512
 827 C                   end                                                   E01       010812  76800 11.11.32.512
 829  *----------------------------------------------------------------------            010812  77000 11.11.32.512
 830  *  %found not%found BIF                                                            010812  77100 11.11.32.512
 832  * NO INDICATOR USED ON CHAIN                                                       010812  77300 11.11.32.512
 833  * AUDIT RPGIV CHAIN STATEMENT (NO ERROR INDICATOR)                                  010812  77400 11.11.32.512
 834 C                   eval      oorder = 1500                                         010812  77500 11.11.32.512
                                     1500
 835 C                   eval      oline = 1                                             010812  77600 11.11.32.513
                                       1
 836 C     ordkey        chain     odetrec                                      25 IS NOT010812  77700 11.11.32.513
           000150000001
ODORD#-0001500 ODLINE-00001 ODCUST-0001000 ODSTOR-0000522 ODITEM-Y1815        ODPRIC-0002100 ODQTY-0000002 ODREQD-20000317
ODEXPD-20070121 ODSHPD-00000000 ODINV#-0000000 ODSTAT-O ODX-
```

```
 837 C                   if        not%found                                   B01       010812  77800 11.11.32.513
 839 C                   END                                                   E01       010812  78000 11.11.32.513
 841 C                   if        %found                                      B01       010812  78200 11.11.32.513
 842 C                   eval      *IN82 = *on                                  01       010812  78300 11.11.32.513
                                   1
 843 C                   END                                                   E01       010812  78400 11.11.32.513
 844  *----------------------------------------------------------------------            010814  78500 11.11.32.513
 846  *-----------------------------------------------------------------                 010814  78700 11.11.32.513
 847  * READ ALL CUSTOMER STORE RECORDS IN THE FILE                                      010814  78800 11.11.32.513
 848  *-----------------------------------------------------------------                 010814  78900 11.11.32.513
 849 C     CUCUST        SETLL     CUSTREC1                                              010814  79000 11.11.32.513
           0002050
 850 C                   READ      CUSTREC1                                              010814  79100 11.11.32.514
CUCUST-0002050  CUSTOR-0000000  CUNAME-XYZ  CORPORATE  OFFICE           CUAD1-555  ARCH  STREET                      CUAD2-
CUSTA-PA
 851 C                   DOW       not %eof(CUSTMAST)                          B01       010814  79200 11.11.32.514
 852 C                   EXCEPT    PRTCUS                                       01       010814  79300 11.11.32.514
 853  * READ ANOTHER RECORD IN THE CUSTOMER FILE                                         010814  79400 11.11.32.514
 854 C                   READ      CUSTREC1                                     01       010814  79500 11.11.32.514
CUCUST-0002050  CUSTOR-0000001  CUNAME-XYZ  STORE  -  ARDMORE          CUAD1-122  MONTGOMERY  AVE            CUAD2-THIRD  FLOOR
CUSTA-PA
 851 C                   DOW       not %eof(CUSTMAST)                          B01       010814  79200 11.11.32.514
 852 C                   EXCEPT    PRTCUS                                       01       010814  79300 11.11.32.514
 853  * READ ANOTHER RECORD IN THE CUSTOMER FILE                                         010814  79400 11.11.32.514
 854 C                   READ      CUSTREC1                                     01       010814  79500 11.11.32.514
 855 C                   ENDDO                                              OF E01       010814  79600 11.11.32.514
 859  *----------------------------------------------------------------------            010812  80000 11.11.32.514
 860  *  %ELEM BIF                                                                       010812  80100 11.11.32.514
 861 C                   Z-ADD     *zero         workcounter      10 2                   010812  80200 11.11.32.514
                                                         .00
 862 C                   eval      Indx = 1                                              010812  80300 11.11.32.514
                                      1
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.514
                                      1

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.514
                                                      123.45
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.515
                                      2      2
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.515
                                      2

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.515
                                                      246.90
                                                                                                       11.10.48.733   PAGE
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.515
                                      3      3
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.515
                                      3

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.515
                                                      370.35
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.515
                                      4      4
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.515
                                      4

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.515
                                                      493.80
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.515
                                      5      5
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                      5

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                      617.25
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
```

```
                                      6      6
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                      6

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                      740.70
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                      7      7
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                      7

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                      864.15
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                      8      8
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                      8

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                      987.60
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                      9      9
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                      9

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                     1111.05
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                     10     10
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                     10

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                     1234.50
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                     11     11
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                     11

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                     1357.95
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                     12     12
 863 C                   DOW       Indx <= %ELEM(LocatTotal)                   B01       010812  80400 11.11.32.516
                                     12

000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
 864 C                   add       123.45        workcounter                    01       010812  80500 11.11.32.516
                                                     1481.40
 865 C                   EVAL      Indx = Indx + 1                              01       010812  80600 11.11.32.516
                                     13     13
 866 C                   ENDDO                                                 E01       010812  80700 11.11.32.520
 868 C                   Z-ADD     45            NEXT              2 0                   010812  80900 11.11.32.520
                                                   45
 869 C                   Z-ADD     *zero         counter                                 010812  81000 11.11.32.520
                                                       0
 870 C                   movel     'W'           CKASTA            1                     010812  81100 11.11.32.520
                                                 W
                                                                                                       11.10.48.733   PAGE
 871 C                   DOW       COUNTER < 6 AND                             B01       010812  81200 11.11.32.520
                                         0
 828 C                             CKASTA = 'W'                                B01       010812  81300 11.11.32.520
                                   W
 873 C                   ADD       1             COUNTER           2 0          01       010812  81400 11.11.32.520
                                                       1
 871 C                   DOW       COUNTER < 6 AND                             B01       010812  81200 11.11.32.520
```

```
                                        1
828 C                             CKASTA = 'W'                                B01       010812  81300 11.11.32.520
                                  W
873 C                   ADD       1             COUNTER           2 0          01       010812  81400 11.11.32.520
                                                      2
871 C                   DOW       COUNTER < 6 AND                             B01       010812  81200 11.11.32.520
                                        2
828 C                             CKASTA = 'W'                                B01       010812  81300 11.11.32.520
                                  W
873 C                   ADD       1             COUNTER           2 0          01       010812  81400 11.11.32.520
                                                      3
871 C                   DOW       COUNTER < 6 AND                             B01       010812  81200 11.11.32.520
                                        3
828 C                             CKASTA = 'W'                                B01       010812  81300 11.11.32.520
                                  W
873 C                   ADD       1             COUNTER           2 0          01       010812  81400 11.11.32.520
                                                      4
871 C                   DOW       COUNTER < 6 AND                             B01       010812  81200 11.11.32.520
                                        4
828 C                             CKASTA = 'W'                                B01       010812  81300 11.11.32.520
                                  W
873 C                   ADD       1             COUNTER           2 0          01       010812  81400 11.11.32.520
                                                      5
871 C                   DOW       COUNTER < 6 AND                             B01       010812  81200 11.11.32.522
                                        5
828 C                             CKASTA = 'W'                                B01       010812  81300 11.11.32.522
                                  W
873 C                   ADD       1             COUNTER           2 0          01       010812  81400 11.11.32.522
                                                      6
874 C                   ENDDO                                                 E01       010812  81500 11.11.32.522
877 C                   movel     '1'           @yes              1                     010812  81800 11.11.32.522
                                                1
878 C                   movel     '1'           @1stline          1                     010812  81900 11.11.32.522
                                                1
879 C                   dou       @1stline = @yes or                          B01       010812  82000 11.11.32.522
836 C                             *IN33 = *OFF                                B01       010812  82100 11.11.32.522
881 C                   movel     '3'           hold2             1            01       010812  82200 11.11.32.522
                                                3
882 C                   enddo                                                 E01       010812  82300 11.11.32.522
883 C                   EVAL      COUNTER = NEXT +                                      010812  82400 11.11.32.522
                                       58     45
840 C                             15 - 2                                                010812  82500 11.11.32.522
885 C                   movel     'E'           CKRTFL            1                     010812  82600 11.11.32.523
                                                E
886 C                   movel     'N'           @no               1                     010812  82700 11.11.32.523
                                                N
887 C                   movel     ' '           @outpt            1                     010812  82800 11.11.32.523
888 C                   movel     'WMAR_EMPL'   L1RPTN            9                     010812  82900 11.11.32.523
                                                WMAR_EMPL
890 C                   select                                                B01       010812  83100 11.11.32.523
891 C                   when      L1RPTN = 'WMAR_EMPL' OR                     X01       010812  83200 11.11.32.523
                                  WMAR_EMPL
892 C                             L1RPTN = 'RTPA_EMPL'                        X01       010812  83300 11.11.32.523
                                  WMAR_EMPL
891 C                   when      L1RPTN = 'WMAR_EMPL' OR                     X01       010812  83200 11.11.32.523
                                  WMAR_EMPL
848 C                             L1RPTN = 'RTPA_EMPL'                        X01       010812  83300 11.11.32.523
                                  WMAR_EMPL
894 C                   IF        (CKASTA = 'W'     OR                        B02       010812  83500 11.11.32.523
                                   W
895 C                              CKASTA = 'F'     OR                        B02       010812  83600 11.11.32.523
                                   W
896 C                              CKASTA = 'X'     OR                        B02       010812  83700 11.11.32.523
                                   W
897 C                              CKRTFL = 'E')                              B02       010812  83800 11.11.32.523
                                   E
898 C                   EVAL      @outpt = @no                                 02       010812  83900 11.11.32.523
                                  N        N
899 C                   ENDIF                                                 E02       010812  84000 11.11.32.523
904 C                   ENDSL                                                 E01       010812  84500 11.11.32.523
                                                                                                      11.10.48.733   PAGE
906 C                   EXSR      stdsubroutine1                                        011129  84700 11.11.32.524
```

```
 908 C                   EXSR      stdsubroutine2                                        011129  84900 11.11.32.524
 909 c                   call      'TEST3   '                                   CALL BATC040624  85000 11.11.32.524
 910  *  %eof BIF                                                                        011211  85100 11.11.32.524
 911 C                   eval      cucust = 1000                                         011211  85200 11.11.32.524
                                     1000
 912 C     CUCUST        SETLL     CUSTREC1                                              011211  85300 11.11.32.525
           0001000
 913 C     CUCUST        READE     CUSTREC1                                              011211  85400 11.11.32.525
           0001000
CUCUST-0001000  CUSTOR-0000000  CUNAME-ABC  STORES  INC.                     CUAD1-15  CORPORATE  DRIVE               CUAD2-
CUSTA-PA
 914 C                   DOW       not %eof(CUSTMAST)                          B01       011211  85500 11.11.32.525
 915 C                   EXCEPT    PRTCUS                                       01       011211  85600 11.11.32.525
 916  * READ ANOTHER RECORD                                                              011211  85700 11.11.32.525
 917 C     CUCUST        READE     CUSTREC1                                     01       011211  85800 11.11.32.525
           0001000
CUCUST-0001000  CUSTOR-0000001  CUNAME-ABC  STORES  INC                        CUAD1-423  MONTGOMERY  AVENUE          CUAD2-
CUSTA-PA
 914 C                   DOW       not %eof(CUSTMAST)                          B01       011211  85500 11.11.32.525
 915 C                   EXCEPT    PRTCUS                                       01       011211  85600 11.11.32.525
 916  * READ ANOTHER RECORD                                                              011211  85700 11.11.32.525
 917 C     CUCUST        READE     CUSTREC1                                     01       011211  85800 11.11.32.525
           0001000
CUCUST-0001000  CUSTOR-0000002  CUNAME-ABC  STORES    STORE  #2           CUAD1-554  ARCH  STREET                     CUAD2-
CUSTA-PA
 914 C                   DOW       not %eof(CUSTMAST)                          B01       011211  85500 11.11.32.525
 915 C                   EXCEPT    PRTCUS                                       01       011211  85600 11.11.32.525
 916  * READ ANOTHER RECORD                                                              011211  85700 11.11.32.525
 917 C     CUCUST        READE     CUSTREC1                                     01       011211  85800 11.11.32.525
           0001000
CUCUST-0001000  CUSTOR-0000522  CUNAME-ABC  STORES    STORE  #522        CUAD1-231  70TH  STREET                      CUAD2-
CUSTA-NY
 914 C                   DOW       not %eof(CUSTMAST)                          B01       011211  85500 11.11.32.525
 915 C                   EXCEPT    PRTCUS                                       01       011211  85600 11.11.32.525
 916  * READ ANOTHER RECORD                                                              011211  85700 11.11.32.525
 917 C     CUCUST        READE     CUSTREC1                                     01       011211  85800 11.11.32.525
           0001000
 918 C                   ENDDO                                              OF E01       011211  85900 11.11.32.525
 920 c                   eval      x = 1                                                 020705  86100 11.11.32.525
                                   1
 921 c                   eval      srt(x) = '  34567   '                                  020705  86200 11.11.32.525
                                     34567
 922 c                   eval      z = 1                                                 020705  86300 11.11.32.525
                                   1
 923 c                   eval      sta(z) = ' BC'                                        020705  86400 11.11.32.525
                                    BC
 925 c                   eval      srt(x) = %trimr(sta(z))                               020705  86600 11.11.32.525
                                    BC
                                                    BC
 927 c                   eval      srt(x) = %trimr(srt(x)) + apos +                      020705  86800 11.11.32.525
                                    BC' BC'
                                                    BC' BC'
 884 c                                      %TRIMR(STA(Z)) + APOS                        020705  86900 11.11.32.525
                                                    BC
 929  *----------------------------------------------------------------------            010812  87000 11.11.32.525
 930 C                   SETON                                            LR             000324  87100 11.11.32.525
 931 C                   RETURN                                                 RETURN TO 000316  87200 11.11.32.525
```

RTPA for RPG Audit Output – BATCHPGM1 (Batch) RPGLE – Called from program NEWEXPSH

Note – RTPA audit output records the parameters passed to and from called programs

```
------------------------------------------------------------------------------------------------------------------------
Program: BATCHPGM1 Batch program with call to     Obj Lib: Z$AUDITE    Initiated:  01/30/07 17.05.42.106   PAGE   1
         BATCHPGM1   BATCHPGM1
Job: 056103              User Profile: PHH                               Source File/Library:  QRPGLESRC Z$AUDIT
Line#                                                                        Do# SrcId  ChgDat  Seq#   Time
   4 C     *ENTRY        PLIST                                                      010529   400
   5 C                   PARM                    PARMA            79                010529   500 17.05.42.126
                                                 AAAAAAAAAAAAAAAAAAAAAAAAAAAAA
   6 C                   PARM                    PARMB            79                010529   600 17.05.42.126
                                                 BBBBBBBBBBBBBBBBBBBBBBBBBBBBB
   8 C                   MOVEL     'AAAAAAAA'    CHECK8            8                010827   800 17.05.42.131
                                                 AAAAAAAA
   9 C                   Z-ADD     5             FIRST             2 0              000521    900 17.05.42.131
                                                     5
  10 C                   Z-ADD     14.2          SECND             3 2              000521   1000 17.05.42.131
                                                  4.20
  11 C     FIRST         MULT      SECND         PROD              5 2              000521   1100 17.05.42.131
              5                    4.20
                                                21.00
  13 C     ANSR          IFNE      15                                    B01       000521   1300 17.05.42.131
              0
  14 C     ANSR          ANDNE     14                                     01       000521   1400 17.05.42.131
              0
  15 C     'AAAAAAAA'    OREQ      CHECK8                                 01       010827   1500 17.05.42.131
                                   AAAAAAAA
  16 C     100           DIV       8             ANSR              3 0    01       000521   1600 17.05.42.131
                                                   12
  17 C                   MVR                     FRACT             2 0    01       000521   1700 17.05.42.131
                                                     4
  18 C                   END                                             E01       000521   1800 17.05.42.131
  20 C     FRACT         IFNE      0                                     B01       000521   2000 17.05.42.132
               4
  21 C                   Z-ADD     FRACT         PRTFLD            2 0    01       000521   2100 17.05.42.132
                                       4
                                                      4
  22 C                   END                                             E01       000521   22800 17.05.42.131
  24 C                   MOVEL     *ALL'C'       @MSGDC                            051007   2400 17.05.42.132
                                                 CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  25 C                   MOVEL     *ALL'D'       @MSGDD                            051007   2500 17.05.42.132
                                                 DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  26 C                   MOVEL     *ALL'E'       @MSGDE                            051007   2600 17.05.42.132
                                                 EEEEEEEEEEEEEEEEEEEEEEEEEEEEE
  27  * call with parms                                                             051007   2700 17.05.42.132
  28 C                   CALL      'BATCHPGM2'                                     051007   2800 17.05.42.132
  29 C                   PARM                    @MSGDC           40               051007   2900 17.05.42.132
                                                 CCCCCCCCCCCCCCCCCCCCCCCCCCCCC
  30 C                   PARM                    @MSGDD           50               051007   3000 17.05.42.132
                                                 DDDDDDDDDDDDDDDDDDDDDDDDDDDDD
  31 C                   PARM                    @MSGDE           60               051007   3100 17.05.42.132
                                                 EEEEEEEEEEEEEEEEEEEEEEEEEEEEE
  32 C                   SETON                                                LR   000324   3200 17.05.42.132
  33 C                   RETURN                                                    000316   3300 17.05.42.132
---------------------------------------------------------------------------------------------------------------
```