

XPEDITER/TSO — Stepping Through Program Execution

General Questions

Question	Page(s)
Can I issue a GO command to the next paragraph? (<i>COBOL only</i>)	2
Can I issue a GO command to the next label? (<i>Assembler only</i>)	2
Can I execute more than one statement at a time?	2
Can I back up after a GO command?	2
Is GO 1 valid in an unattended batch test?	2
What is GO HALF?	2
How can I set up my PF keys to step through the program?	3

Functionality / Scenario Questions

Question	Page(s)
How can I debug a called program if both the calling and called programs are compiled with XPEDITER?	4
How can I debug a called program if only the called program is compiled with XPEDITER?	9
How can I debug a called program if the calling program does not yet exist?	15

NOTE: The following examples apply to COBOL, PL/I, C and Assembler unless otherwise indicated.

NOTE: For more information about stepping through your program when certain conditions are met, see the Rapid response section Gaining Control Conditionally.

Can I issue a GO command to the next paragraph? (*COBOL only*)

You can issue a GO command to the next paragraph by using the GO PARA command. You can also use the GO *n* PARA and go until the next *n* paragraph is reached.

Can I issue a GO command to the next label? (*Assembler only*)

You can issue a GO command to the next label by using the GO LABEL command. You can also use the GO *n* LABEL and go until the next *n* label is reached.

Can I execute more than one statement at a time?

You can execute more than one statement at a time by using the GO *n* command. GO *n* TRACE executes and traces the next *n* statements or labeled statements.

Can I back up after a GO command?

You can back up after a GO command if you use the MONITOR command before you begin executing your program. When you want to back up, use the REVERSE command or use the GOTO command to reset the execution to a previous statement. For more information, see the Rapid Response document called *Reviewing Program Execution*.

Is GO 1 valid in an unattended batch test?

No, GO 1 is not valid in an unattended batch test.

What is GO HALF?

GO HALF executes a half step and then pauses. If you are positioned before a statement that has executed, GO HALF will execute the statement and stop after the execution. The execution pointer remains on that statement but changes from a single-head arrow (before) to a double-head arrow (after). If you are positioned after a statement that has executed, the execution pointer has a double-head arrow. GO HALF will take you to the next statement and stop before it executes.

How can I set up my PF keys to step through the program?

The GO 1 command is mapped to the PF9 and PF21 keys. The GO command is mapped to the PF12 and PF24 keys. This is a complete list of the default PF keys.

PF1	HELP	PF13	HELP
PF2	PEEK CSR	PF14	FIND CSR
PF3	END	PF15	END
PF4	EXIT	PF16	EXIT
PF5	FIND	PF17	FIND IND
PF6	LOCATE *	PF18	LOCATE *
PF7	UP	PF19	UP
PF8	DOWN	PF20	DOWN
PF9	GO 1	PF21	GO 1
PF10	LEFT	PF22	DLEFT
PF11	RIGHT	PF23	DRIGHT
PF12	GO	PF24	GO

You can also change the default PF keys. From the XPEDITER/TSO Primary menu, select option 1 (Defaults) and then option 3 (Test PF Keys) from the Defaults menu. On the XPEDITER PF Key Definitions and Labels screen, you can change the functions of the PF keys.

How can I debug a called program if both the calling and called programs are compiled with XPEDITER?

1. Begin a new test of your program.

```
Profile: DEFAULT ----- XPEDITER/TSO - STANDARD (2.1) -----
COMMAND ===> [ENTER]

COMMANDS:  Setup (Display Setup Menu)

TEST SELECTION CRITERIA:

          Program ===> CWXTCOB
          Entry Point ===>
          Load Module ===>

          Initial Script ===>
          Post Script ===>

          PARM String ===> 00003

File List/JCL Member ===> 'USERID.XPEDITER.V6R5M0.SAMPLIB(CWXTJCLC) '

Code Coverage Test? ===> NO
Is This a DB2 Test? ===> NO   Plan ===>           System ===>
Test with XPEDITER+ ===> NO   Qualified LU name ===> .
          Press ENTER to process or enter END command to terminate
```

Figure 1: Test Standard Screen

INTERCEPT Display the source for a program and set the before and after module breakpoints.

Primary Commands: INTERCEPT, INT

Example: INT CWXTSUBC

2. Intercept program CWXTSUBC.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> intercept cwxtsubc                                SCROLL ==> CSR
                BEFORE BREAKPOINT ENCOUNTERED
                -----
000347  01 PARMINFO                                         >  ..00003
                ** END **

----- Before CWXTCOB <> -----
====> B  PROCEDURE DIVISION USING PARMINFO.
000351      0000-MAINLINE.
000352          PERFORM 9000-OPEN.
000353          PERFORM 9100-CHECK-PARM.
000354          IF GOOD-PARM
000355              PERFORM 9200-INIT
000356              PERFORM 8000-READ-INPUT
000357                  UNTIL RECORDS-READ = START-NUMBER
000358                  OR END-OF-FILE
000359          PERFORM 1000-PROCESS-DATA
000360              UNTIL END-OF-FILE
000361          PERFORM 6000-PRINT-EMPLOYEE-REPORT
000362          MOVE 1 TO REGION-SUB

```

Figure 2: INTERCEPT Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 02/25/99  COMP TIME: 09:47:00
                ** END **

----- Before CWXTCOB <> -----
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
000073 A      GOBACK.
000074          ***
000075          ***
000076          ***
000077          1000-PROCESS-SALES.

```

Figure 3: Results of INTERCEPT Command

3. Modify the program field to display the source for CWXTCOB.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
PROGRAM: cwxtcob  MODULE: CWXTCOB  COMP DATE: 02/25/99  COMP TIME: 09:47:00
** END **

----- Before CWXTCOB <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
000073 A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1

```

Overtyping the program name.

Figure 4: Overtyping Program Field to Display Source for CWXTCOB

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
PROGRAM: CWXTCOB  MODULE: CWXTCOB  COMP DATE: 02/25/99  COMP TIME: 09:47:36
** END **

----- Before CWXTCOB <>
=====> B  PROCEDURE DIVISION USING PARMINFO.
000351      0000-MAINLINE.
000352          PERFORM 9000-OPEN.
000353          PERFORM 9100-CHECK-PARM.
000354          IF GOOD-PARM
000355              PERFORM 9200-INIT
000356              PERFORM 8000-READ-INPUT
000357                  UNTIL RECORDS-READ = START-NUMBER
000358                  OR END-OF-FILE
000359              PERFORM 1000-PROCESS-DATA
000360                  UNTIL END-OF-FILE
000361              PERFORM 6000-PRINT-EMPLOYEE-REPORT
000362              MOVE 1 TO REGION-SUB
000363              IF END-OF-MONTH
000364              PERFORM 7000-PRINT-REGION-REPORT

```

Figure 5: Source for CWXTCOB

4. Begin program execution.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> go                                SCROLL ==> CSR
PROGRAM: CWXTCOB      MODULE: CWXTCOB  COMP DATE: 02/25/99  COMP TIME: 09:47:36
                                -----
000347  01 PARMINFO                >  ..00003
      ** END **

----- Before CWXTCOB <>
===== B  PROCEDURE DIVISION USING PARMINFO.
000351  0000-MAINLINE.
000352      PERFORM 9000-OPEN.
000353      PERFORM 9100-CHECK-PARM.
000354      IF GOOD-PARM
000355          PERFORM 9200-INIT
000356          PERFORM 8000-READ-INPUT
000357              UNTIL RECORDS-READ = START-NUMBER
000358              OR END-OF-FILE
000359          PERFORM 1000-PROCESS-DATA
000360              UNTIL END-OF-FILE
000361          PERFORM 6000-PRINT-EMPLOYEE-REPORT
000362          MOVE 1 TO REGION-SUB
000363          IF END-OF-MONTH
000364          PERFORM 7000-PRINT-REGION-REPORT
    
```

Figure 6: GO Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
                                BEFORE BREAKPOINT ENCOUNTERED
000060  01 EMP-TYPE                >  S
000061  01 SALES-AMOUNT            >  055000.00          DECIMAL
000062  01 COMM-TOTAL              >  00000.00          PACKED
      ** END **

----- Before CWXTSUBC <>
===== B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067  0000-MAINLINE.
000068      IF EMP-TYPE EQUAL 'S'
000069          PERFORM 1000-PROCESS-SALES
000070      ELSE
000071          IF EMP-TYPE EQUAL 'M'
000072              PERFORM 2000-PROCESS-MANAGEMENT.
000073 A  GOBACK.
000074      ***
000075      ***
000076      ***
000077  1000-PROCESS-SALES.
000078      MOVE SALES-AMOUNT TO SALES-RANGE.
000079      IF S-RANGE1
000080          SET SALES-IX TO 1
    
```

The cursor is now positioned at the BEFORE breakpoint on the subroutine.

Figure 7: Results of GO Command

5. Exit the session.

```
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> exit                                SCROLL ==> CSR
                BEFORE BREAKPOINT ENCOUNTERED
                -
000060  01 EMP-TYPE                               > S
000061  01 SALES-AMOUNT                           > 055000.00          DECIMAL
000062  01 COMM-TOTAL                             > 00000.00          PACKED
                ** END **

----- Before CWXTSUBC <>
====> B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067  0000-MAINLINE.
000068      IF EMP-TYPE EQUAL 'S'
000069          PERFORM 1000-PROCESS-SALES
000070      ELSE
000071          IF EMP-TYPE EQUAL 'M'
000072              PERFORM 2000-PROCESS-MANAGEMENT.
000073 A  GOBACK.
000074      ***
000075      ***
000076      ***
000077  1000-PROCESS-SALES.
000078      MOVE SALES-AMOUNT TO SALES-RANGE.
000079      IF S-RANGE1
000080          SET SALES-IX TO 1
```

Figure 8: EXIT

How can I debug a called program if only the called program is compiled with XPEDITER?

NOTE: This scenario assumes a DDIO source listing member exists for CWXTSUBC but not for CWXTCOB.

1. Begin a new test of your program.

```

Profile: DEFAULT ----- XPEDITER/TSO - STANDARD (2.1) -----
COMMAND ==> [ENTER]

COMMANDS:  Setup (Display Setup Menu)

TEST SELECTION CRITERIA:

          Program ==> CWXTCOB
          Entry Point ==>
          Load Module ==>

          Initial Script ==>
          Post Script ==>

          PARM String ==> 00003

          File List/JCL Member ==> 'USERID.XPEDITER.V6R5M0.SAMPLIB(CWXTJCLC)'
```

```

Code Coverage Test? ==> NO
Is This a DB2 Test? ==> NO      Plan ==>          System ==>
Test with XPEDITER+ ==> NO      Qualified LU name ==>
          Press ENTER to process or enter END command to terminate
```

Figure 1: Test Standard Screen

```

----- XPEDITER/TSO - LOG -----
COMMAND ==>
NO SOURCE LISTING DATA SET MEMBER FOR CWXTCOB
----- Before CWXTCOB ->
***** TOP OF DATA *****
+-----+
:                                     JOB: USERID      :
: XPEDITER/TSO RELEASE 06.05.00      CUSTOMER # 010000  STEP: ATSOPROC   :
:                                     DATE: 05/27/1999   :
: COMPUWARE MIS PRODUCTION          TIME: 10.13.14    :
:                                     :
+-----+

XPED TSO SPF
TEST CWXTCOB
PARM '00003'
*** CWXTCOB FROM USERID.LOAD          LINK 06/19/98
NO SOURCE LISTING INFORMATION FOUND FOR MODULE
***** BOTTOM OF DATA *****
```

Figure 2: No Source Listing Exists for Program CWXTCOB

2. Intercept program CWXTSUBC.

```

----- XPEDITER/TSO - LOG -----
COMMAND ==> intercept cwxtsubc                                SCROLL ==> CSR
NO SOURCE LISTING DATA SET MEMBER FOR CWXTCOB
----- Before CWXTCOB ->
***** TOP OF DATA *****
+-----+
:                                     JOB: USERID           :
: XPEDITER/TSO RELEASE 06.05.00      CUSTOMER # 010000      STEP: ATSOPROC      :
:                                     DATE: 05/27/1999          :
: COMPUWARE MIS PRODUCTION           TIME: 10.13.14       :
:                                     :                   :
+-----+

XPED TSO SPF
TEST CWXTCOB
PARM '00003'
*** CWXTCOB FROM USERID.LOAD                                LINK 06/19/98
NO SOURCE LISTING INFORMATION FOUND FOR MODULE
***** BOTTOM OF DATA *****
    
```

Figure 3: INTERCEPT Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>
PROGRAM: CWXTSUBC      MODULE: CWXTCOB   COMP DATE: 06/19/98  COMP TIME: 09:45:35
** END **

----- Before CWXTCOB <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
000073 A  GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1
    
```

The source for the subroutine is displayed and it is set before and after breakpoints at the beginning and end of the program.

Figure 4: Result of INTERCEPT Command

3. Begin execution of the driver program, CWXTCOB.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> go                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 06/19/98  COMP TIME: 09:45:35
      ** END **

----- Before CWXTCOB <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
000073 A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1
    
```

Figure 5: GO Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
                                BEFORE BREAKPOINT ENCOUNTERED
000060      01 EMP-TYPE                    > S
000061      01 SALES-AMOUNT                > 055000.00          DECIMAL
000062      01 COMM-TOTAL                  > 000000.00          PACKED
      ** END **

----- Before CWXTSUBC <>
===== B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
000073 A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1
    
```

Execution stopped at the BEFORE breakpoint at the beginning of the subroutine.

Figure 6: Results of GO Command

4. Resume execution of the subroutine.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> go                                SCROLL ==> CSR
                BEFORE BREAKPOINT ENCOUNTERED
                -
000060  01 EMP-TYPE                            > S
000061  01 SALES-AMOUNT                        > 055000.00          DECIMAL
000062  01 COMM-TOTAL                          > 00000.00          PACKED
                ** END **

----- Before CWXTSUBC <>
====> B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067  0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069          PERFORM 1000-PROCESS-SALES
000070          ELSE
000071          IF EMP-TYPE EQUAL 'M'
000072          PERFORM 2000-PROCESS-MANAGEMENT.
000073 A  GOBACK.
000074  ***
000075  ***
000076  ***
000077  1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080          SET SALES-IX TO 1

```

Figure 7: GO Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 06/19/98  COMP TIME: 09:45:35
                ** END **

----- After CWXTSUBC <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067  0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069          PERFORM 1000-PROCESS-SALES
000070          ELSE
000071          IF EMP-TYPE EQUAL 'M'
000072          PERFORM 2000-PROCESS-MANAGEMENT.
====>> A  GOBACK.
000074  ***
000075  ***
000076  ***
000077  1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080          SET SALES-IX TO 1

```

Figure 8: Results of GO Command

5. Resume program execution. The main program re-executes the driver program multiple times, so it will return once again to the procedure division.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> go                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 02/25/99  COMP TIME: 09:47:36
      ** END **

-----
----- After CWXTSUBC <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
====> A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1

```

Figure 9: GO Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
                                BEFORE BREAKPOINT ENCOUNTERED
000060      01 EMP-TYPE                    > S
000061      01 SALES-AMOUNT                > 070000.00          DECIMAL
000062      01 COMM-TOTAL                  > 00000.00          PACKED
      ** END **

-----
----- Before CWXTSUBC <>
====> B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
000073 A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1

```

Figure 10: Results of GO Command

6. Exit the session.

```
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> exit                                SCROLL ==> CSR
                BEFORE BREAKPOINT ENCOUNTERED
                -
000060  01 EMP-TYPE                               > S
000061  01 SALES-AMOUNT                           > 070000.00          DECIMAL
000062  01 COMM-TOTAL                             > 00000.00          PACKED
                ** END **
----- Before CWXTSUBC <>
====> B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067  0000-MAINLINE.
000068      IF EMP-TYPE EQUAL 'S'
000069          PERFORM 1000-PROCESS-SALES
000070      ELSE
000071          IF EMP-TYPE EQUAL 'M'
000072              PERFORM 2000-PROCESS-MANAGEMENT.
000073 A  GOBACK.
000074      ***
000075      ***
000076      ***
000077  1000-PROCESS-SALES.
000078      MOVE SALES-AMOUNT TO SALES-RANGE.
000079      IF S-RANGE1
000080          SET SALES-IX TO 1
```

Figure 11: EXIT

How can I debug a called program if the calling program does not yet exist?

1. Begin a new test of your program - no parm string or File List/JCL Member is required.

```
Profile: DEFAULT ----- XPEDITER/TSO - STANDARD (2.1)
COMMAND ===> [ENTER]

COMMANDS:  Setup (Display Setup Menu)

TEST SELECTION CRITERIA:

          Program ===> CWXTSUBC
          Entry Point ===>
          Load Module ===>

          Initial Script ===>
          Post Script ===>

          PARM String ===>

          File List/JCL Member ===>

          Code Coverage Test? ===> NO
          Is This a DB2 Test? ===> NO      Plan ===>          System ===>
          Test with XPEDITER+ ===> NO      Qualified LU name ===> .
          Press ENTER to process or enter END command to terminate
```

Figure 1: Test Standard Screen

2. Change EMP-TYPE to 'S' and SALES-AMOUNT to 2000.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
                                BEFORE BREAKPOINT ENCOUNTERED
000060 01 EMP-TYPE                        > -
000061 01 SALES-AMOUNT                    > 002000.00          DECIMAL
000062 01 COMM-TOTAL                      > 00000.00          PACKED
** END **

----- Before CWXTSUBC <>
====> B PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067 0000-MAINLINE.
000068     IF EMP-TYPE EQUAL 'S'
000069     PERFORM 1000-PROCESS-SALES
000070     ELSE
000071     IF EMP-TYPE EQUAL 'M'
000072     PERFORM 2000-PROCESS-MANAGEMENT.
000073 A GOBACK.
000074 ***
000075 ***
000076 ***
000077 1000-PROCESS-SALES.
000078     MOVE SALES-AMOUNT TO SALES-RANGE.
000079     IF S-RANGE1
000080     SET SALES-IX TO 1
    
```

Overtyping the data here.

Figure 2: Overtyping Values

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTSUBC  COMP DATE: 06/19/98  COMP TIME: 09:45:35
                                -
000060 01 EMP-TYPE                        > S
000061 01 SALES-AMOUNT                    > 002000.00          DECIMAL
000062 01 COMM-TOTAL                      > 00000.00          PACKED
** END **

----- Before CWXTSUBC <>
====> B PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067 0000-MAINLINE.
000068     IF EMP-TYPE EQUAL 'S'
000069     PERFORM 1000-PROCESS-SALES
000070     ELSE
000071     IF EMP-TYPE EQUAL 'M'
000072     PERFORM 2000-PROCESS-MANAGEMENT.
000073 A GOBACK.
000074 ***
000075 ***
000076 ***
000077 1000-PROCESS-SALES.
000078     MOVE SALES-AMOUNT TO SALES-RANGE.
000079     IF S-RANGE1
000080     SET SALES-IX TO 1
    
```

Figure 3: Results of Overtyping Values

3. Begin program execution.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> go                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTSUBC  COMP DATE: 06/19/98  COMP TIME: 09:45:35
-
000060  01 EMP-TYPE                               > S
000061  01 SALES-AMOUNT                           > 002000.00          DECIMAL
000062  01 COMM-TOTAL                             > 00000.00          PACKED
** END **

----- Before CWXTSUBC <>
====> B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067  0000-MAINLINE.
000068      IF EMP-TYPE EQUAL 'S'
000069      PERFORM 1000-PROCESS-SALES
000070      ELSE
000071      IF EMP-TYPE EQUAL 'M'
000072      PERFORM 2000-PROCESS-MANAGEMENT.
000073 A  GOBACK.
000074  ***
000075  ***
000076  ***
000077  1000-PROCESS-SALES.
000078      MOVE SALES-AMOUNT TO SALES-RANGE.
000079      IF S-RANGE1
000080      SET SALES-IX TO 1

```

Figure 4: GO Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
                                TEST COMPLETED
** END **

----- After CWXTSUBC <>
====> A  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000066 B  0000-MAINLINE.
000067      IF EMP-TYPE EQUAL 'S'
000068      PERFORM 1000-PROCESS-SALES
000069      ELSE
000070      IF EMP-TYPE EQUAL 'M'
000071      PERFORM 2000-PROCESS-MANAGEMENT.
000072
000073 A  GOBACK.
000074  ***
000075  ***
000076  ***
000077  1000-PROCESS-SALES.
000078      MOVE SALES-AMOUNT TO SALES-RANGE.
000079      IF S-RANGE1
000080      SET SALES-IX TO 1

```

Figure 5: Results of GO Command

KEEP LINKAGE Display the linkage section in a keep window.

Primary Commands: **KEEP LINKAGE, K LINKAGE**

4. Keep the linkage section.

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==> keep linkage                                SCROLL ==> CSR
                                TEST COMPLETED
                                ** END **

----- After CWXTSUBC <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
====>> A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1
000080              SET SALES-IX TO 1

```

Figure 6: KEEP Linkage Command

```

----- XPEDITER/TSO - SOURCE -----
COMMAND ==>                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTSUBC  COMP DATE: 06/19/98  COMP TIME: 09:45:35
-
000060 K 01 EMP-TYPE                                > S
000061 K 01 SALES-AMOUNT                            > 002000.00          DECIMAL
000062 K 01 COMM-TOTAL                              > 00040.00           PACKED
                                ** END **

----- After CWXTSUBC <>
000066 B  PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067      0000-MAINLINE.
000068          IF EMP-TYPE EQUAL 'S'
000069              PERFORM 1000-PROCESS-SALES
000070          ELSE
000071              IF EMP-TYPE EQUAL 'M'
000072                  PERFORM 2000-PROCESS-MANAGEMENT.
====>> A      GOBACK.
000074      ***
000075      ***
000076      ***
000077      1000-PROCESS-SALES.
000078          MOVE SALES-AMOUNT TO SALES-RANGE.
000079          IF S-RANGE1

```

Figure 7: Results of KEEP Linkage Command

5. Exit the session.

```
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> exit                                SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTSUBC  COMP DATE: 06/19/98  COMP TIME: 09:45:35
-
000060 K 01 EMP-TYPE                               > S
000061 K 01 SALES-AMOUNT                           > 002000.00          DECIMAL
000062 K 01 COMM-TOTAL                             > 00040.00          PACKED
** END **
----- After CWXTSUBC <>
000066 B PROCEDURE DIVISION USING EMP-TYPE, SALES-AMOUNT, COMM-TOTAL.
000067     0000-MAINLINE.
000068         IF EMP-TYPE EQUAL 'S'
000069             PERFORM 1000-PROCESS-SALES
000070         ELSE
000071             IF EMP-TYPE EQUAL 'M'
000072                 PERFORM 2000-PROCESS-MANAGEMENT.
====>> A     GOBACK.
000074     ***
000075     ***
000076     ***
000077     1000-PROCESS-SALES.
000078         MOVE SALES-AMOUNT TO SALES-RANGE.
000079         IF S-RANGE1
000080             SET SALES-IX TO 1
```

Figure 8: EXIT

