

Xpediter Sourceless Debugging

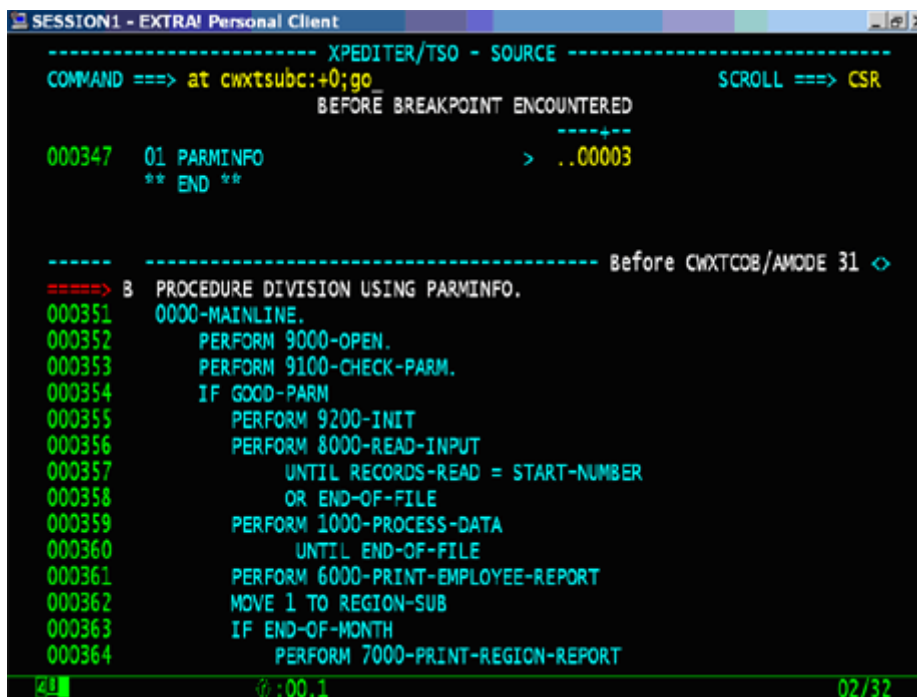
The typical Xpediter test begins by compiling your application program with the Compuware language processor. This process creates a source listing that allows you to interactively step through your source code as it executes. But can you still debug your program with Xpediter if a source listing is not available? Yes you can – using a form of testing called sourceless debugging.

Two methods of sourceless debugging are available. You can choose Xpediter’s traditional support for debugging programs without source or you can choose to have Xpediter create a “pseudo-assembler” view of the object code. You can still use Xpediter to set breakpoints, intercept abends, step through instructions and modify data while testing your program. The primary difference is that no source is available to map to your object code. Therefore, you work with offsets and instructions instead of data names and source code. Test session setup and startup is the same as setting up a session to interactively debug at the source level.

Using Xpediter’s traditional sourceless debugging support, you must use the AT command to set a breakpoint. Enter

AT module name:+offset to set a breakpoint upon entry into the subprogram and begin program execution.

When the AT breakpoint is reached, the AT Display screen shows your sourceless program in dump format.



```
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> at cwxtsubc:+0;go          SCROLL ==> CSR
                BEFORE BREAKPOINT ENCOUNTERED
000347  01 PARMINFO          > ..00003
        ** END **

----- Before CWXTCOB/AMODE 31 <>
===== 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
```

Some Xpediter commands available during a traditional sourceless debugging test session are GPREGS to display the current contents of the general purpose registers, MEMORY to view storage and SHOW ACTIVE to display a summary of register contents, the PSW and some control block information.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - AT DISPLAY -----
COMMAND ==> -                               SCROLL ==> CSR
          AT ADDRESS 375EA6E8 : 47F0F028      : B 40(,15)
----- Before CWXTSUBC/AMODE 31 -----
BASE = 375EA6E8 0 - 2 - 4 - 6 - 8 - A - C - E - = 0-2-4-6-8-A-C-E-
***** TOP OF DATA *****
000000 ==> 00F0F028 00c3c5c5 00000110 00000014 = .00..CEE.....
000010 ==> 47F0F001 98CEAC00 375EA79E 00000000 = .00.....;....
000020 ==> 00000000 00000000 90ECD00C 4110F038 = .....}...0.
000030 ==> 98EFF04C 07FF0000 375EA6E8 00000000 = ..0<.....;Y...
000040 ==> 375EADE0 375EA796 375EA6E8 0012FA38 = ;[\.;...;Y...
000050 ==> 375E9888 375EA782 00104001 00000008 = ;;.....;....
000060 ==> C3E6E7E3 E2E4C2C3 F2F0F0F6 F0F8F1F4 = CWXTSUBC20060814
000070 ==> F1F1F1F8 F2F1F0F3 F0F1F0F1 04740000 = 111821030101...
000080 ==> 0000076C A0687C4C 20000000 00800109 = ...%.@<.....
000090 ==> 01000000 08000000 00000400 00000017 = .....
0000A0 ==> 00000018 00008000 40404040 0008c3E6 = ..... ..CW
0000B0 ==> E7E3E2E4 C2c30500 0001375E 9AD80000 = XTUBC.....;Q..
0000C0 ==> 0000FFFF FFB2375E A6E8375E A7200000 = .....;Y.;....
0000D0 ==> 00080000 0006375E A720375E 9AD8375E = .....;...;Q;
0000E0 ==> 9AB00000 00050000 00000000 00000000 = .....
0000F0 ==> 00000000 00000000 00000000 00000001 = .....
000100 ==> 40404040 40404040 40404040 40404040 = .....
000110 ==> 40404040 40404040 40404040 40400000 = ..
:00.2 02/15

```

Entering the PSEUDOSOURCE command will create and display a “pseudo-assembler” view of your source on the Xpediter source screen. This feature enables you to debug main programs and subprograms in much the same way as with Xpediter's interactive source level debugging. Enter PSEUDOSOURCE module name: to display pseudosource for the module you wish to debug.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - AT DISPLAY -----
COMMAND ==> ps cwxtsubc:                     SCROLL ==> CSR
          AT ADDRESS 375EA6E8 : 47F0F028      : B 40(,15)
----- Before CWXTSUBC/AMODE 31 -----
BASE = 375EA6E8 0 - 2 - 4 - 6 - 8 - A - C - E - = 0-2-4-6-8-A-C-E-
***** TOP OF DATA *****
000000 ==> 00F0F028 00c3c5c5 00000110 00000014 = .00..CEE.....
000010 ==> 47F0F001 98CEAC00 375EA79E 00000000 = .00.....;....
000020 ==> 00000000 00000000 90ECD00C 4110F038 = .....}...0.
000030 ==> 98EFF04C 07FF0000 375EA6E8 00000000 = ..0<.....;Y...
000040 ==> 375EADE0 375EA796 375EA6E8 0012FA38 = ;[\.;...;Y...
000050 ==> 375E9888 375EA782 00104001 00000008 = ;;.....;....
000060 ==> C3E6E7E3 E2E4C2C3 F2F0F0F6 F0F8F1F4 = CWXTSUBC20060814
000070 ==> F1F1F1F8 F2F1F0F3 F0F1F0F1 04740000 = 111821030101...
000080 ==> 0000076C A0687C4C 20000000 00800109 = ...%.@<.....
000090 ==> 01000000 08000000 00000400 00000017 = .....
0000A0 ==> 00000018 00008000 40404040 0008c3E6 = ..... ..CW
0000B0 ==> E7E3E2E4 C2c30500 0001375E 9AD80000 = XTUBC.....;Q..
0000C0 ==> 0000FFFF FFB2375E A6E8375E A7200000 = .....;Y.;....
0000D0 ==> 00080000 0006375E A720375E 9AD8375E = .....;...;Q;
0000E0 ==> 9AB00000 00050000 00000000 00000000 = .....
0000F0 ==> 00000000 00000000 00000000 00000001 = .....
000100 ==> 40404040 40404040 40404040 40404040 = .....
000110 ==> 40404040 40404040 40404040 40400000 = ..
:00.2 02/27

```

You can start debugging using most of the commands you would normally use for a program with a matching source listing. Use the FIND command to locate a particular character string or variable. Enter FIND R6 ALL to locate all references to Register 6.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> f r6 a11_                               SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 08/14/2006  COMP TIME:11:18:21
** END **

----- Before CWXTSUBC/AMODE 31 ->
***** TOP OF MODULE *****
000002          CWXTSUBC  CSECT
000003      47F0 F028      B      40(,R15)          *.00.*
000004          DC      XL8'00c3c5c500000110'      *.CEE...*
000005          DC      XL8'0000001447F0F001'      *....00.*
000006          DC      XL4'98CEAC00'              *q...*
000007      375BF79E      DC      A(CWXTSUBC+X'B6')
000008          DC      12X'00'
000009      90EC D00C      STM     R14,R12,12(R13)    *.}.*
000010      4110 F038      LA      R1,56(,R15)          *.0.*
000011      98EF F04C      LM      R14,R15,76(R15)        *q.0<*
000012      07FF          BR      R15              *..*
000013          DC      XL2'0000'                  *..*
000014      375BF6E8      DC      A(CWXTSUBC)
000015          DC      XL4'0000000'                *...*

```

Every line referencing R6 will be highlighted in the pseudo source display. Primary and line commands can both be used, including BEFORE, AFTER, GO, PEEK, KEEP, COUNT and WHEN. Enter B on line 132 to set a before breakpoint on statement 132 and the primary command GO to begin program execution.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> go_                                       SCROLL ==> CSR
                                18 CHARS 'R6' FOUND
** END **

----- Before CWXTSUBC/AMODE 31 ->
000127      5870 9138      L      R7,312(,R9)          *.}.*
000128      5860 9134      L      R6,308(,R9)          *..*
000129      5820 1000      L      R2,0(,R1)           *..*
000130      4120 2000      LA      R2,0(,R2)           *..*
000131      5020 9134      ST      R2,308(,R9)         *&.j.*
b 132      1862          LR      R6,R2                *..*
000133      5820 1004      L      R2,4(,R1)           *..*
000134      4120 2000      LA      R2,0(,R2)           *..*
000135      5020 9138      ST      R2,312(,R9)         *&.j.*
000136      1872          LR      R7,R2                *..*
000137      5820 1008      L      R2,8(,R1)           *..*
000138      4120 2000      LA      R2,0(,R2)           *..*
000139      5020 913C      ST      R2,316(,R9)         *&.j.*
000140      D203 D088 A034 MVC     136(4,R13),52(R10)  *K.)h..*
000141      BF2F 9158      ICM     R2,15,344(R9)       *.}.*

```

Execution is paused on statement 132 before the load register instruction has executed. As with Xpediter's interactive source level debugging, you can set a permanent display using the KEEP command. Enter K R2;K R6 to display the contents of registers 2 and 6 in the keep window.

```

SESSION1 - EXTRA Personal Client
-----XPEDITER/TSO - SOURCE-----
COMMAND ==> k r2; k r6_          SCROLL ==> CSR
          ** END **
          BEFORE BREAKPOINT ENCOUNTERED

----- Before CWXTSUBC:132/AMODE 31 ->
000127  5870 9138      L   R7,312(R9)  *..j.*
000128  5860 9134      L   R6,308(R9)  *..j.*
000129  5820 1000      L   R2,0(R1)   *...*
000130  4120 2000      LA  R2,0(R2)   *...*
000131  5020 9134      ST  R2,308(R9) *&.j.*
===== B 1862      LR  R6,R2      *..*
000133  5820 1004      L   R2,4(R1)   *...*
000134  4120 2000      LA  R2,0(R2)   *...*
000135  5020 9138      ST  R2,312(R9) *&.j.*
000136  1872          LR  R7,R2      *..*
000137  5820 1008      L   R2,8(R1)   *...*
000138  4120 2000      LA  R2,0(R2)   *...*
000139  5020 913C      ST  R2,316(R9) *&.j.*
000140  D203 D088 A034  MVC 136(4,R13),52(R10) *K.)h.*
000141  BF2F 9158      ICM R2,15,344(R9) *..j.*

```

The contents of Register 2 and Register 6 are displayed in the KEEP window. Enter GO HALF to execute the load register instruction. Whenever a breakpoint is encountered, the keep window is updated to reflect the current values.

```

SESSION1 - EXTRA Personal Client
-----XPEDITER/TSO - SOURCE-----
COMMAND ==> go half          SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 08/14/2006  COMP TIME:11:18:21

SYSTEM K R2                > 3789F3D0
SYSTEM K R6                > 00000000
          ** END **
          BEFORE BREAKPOINT ENCOUNTERED

----- Before CWXTSUBC:132/AMODE 31 ->
000127  5870 9138      L   R7,312(R9)  *..j.*
000128  5860 9134      L   R6,308(R9)  *..j.*
000129  5820 1000      L   R2,0(R1)   *...*
000130  4120 2000      LA  R2,0(R2)   *...*
000131  5020 9134      ST  R2,308(R9) *&.j.*
===== B 1862      LR  R6,R2      *..*
000133  5820 1004      L   R2,4(R1)   *...*
000134  4120 2000      LA  R2,0(R2)   *...*
000135  5020 9138      ST  R2,312(R9) *&.j.*
000136  1872          LR  R7,R2      *..*
000137  5820 1008      L   R2,8(R1)   *...*
000138  4120 2000      LA  R2,0(R2)   *...*
000139  5020 913C      ST  R2,316(R9) *&.j.*
000140  D203 D088 A034  MVC 136(4,R13),52(R10) *K.)h.*
000141  BF2F 9158      ICM R2,15,344(R9) *..j.*

```

The execution pointer will change from a single arrow-head to a double arrow-head, indicating execution is paused after the load register instruction and the keep window has been updated to show the contents of register 2 have been loaded into register 6. Use the MEMORY command to display the data located at the address contained in register 6.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> memory 3789f3d0          SCROLL ==> CSR
                NEXT LOGICAL INSTRUCTION IS CWXTSUBC:133

SYSTEM K R2                > 3789F3D0

SYSTEM K R6                > 3789F3D0
** END **

----- After CWXTSUBC:132/AMODE 31 -->
000127 5870 9138      L   R7,312(R9)      *..j.*
000128 5860 9134      L   R6,308(R9)      *..j.*
000129 5820 1000      L   R2,0(R1)        *...*
000130 4120 2000      LA  R2,0(R2)        *...*
000131 5020 9134      ST  R2,308(R9)      *&.j.*
====>> B 1862        LR  R6,R2          *..*
000133 5820 1004      L   R2,4(R1)        *...*
000134 4120 2000      LA  R2,0(R2)        *...*
000135 5020 9138      ST  R2,312(R9)      *&.j.*
000136 1872          LR  R7,R2          *..*
000137 5820 1008      L   R2,8(R1)        *...*
000138 4120 2000      LA  R2,0(R2)        *...*
000139 5020 913c      ST  R2,316(R9)      *&.j.*
000140 D203 D088 A034   MVC 136(4,R13),52(R10) *K.)h.*
000141 Bf2F 9158        ICM R2,15,344(R9)   *..j.*

```

Memory beginning at address 3789F3D0 is displayed. Enter LOCATE * or hit PF6 to return to the pseudo source display.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - MEMORY -----
COMMAND ==> locate *          SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 08/14/2006  COMP TIME:11:18:21
                After CWXTSUBC:132/AMODE 31 --
BASE = 3789F3D0 0 - 2 - 4 - 6 - 8 - A - C - E - = 0-2-4-6-8-A-C-E-
***** TOP OF DATA *****
000000 ==> E2F0F5F5 F0F0F0F0 F0000000 0F000000 = S05500000.....
000010 ==> 0108300F 0108300F 00000000 00000000 = .....
000020 ==> 00010830 0F000000 000F0000 00000000 = .....
000030 ==> 0000000F 00000000 F0F2F3F3 F4E2F1D9 = .....02334s1R
000040 ==> D6E840C7 C1D9C4D5 C5D94040 4040F3F2 = OY GARDNER 32
000050 ==> F340D6C1 D240E2E3 D9C5C5E3 40C5C4C9 = 3 OAK STREET EDI
000060 ==> D5C14040 40D4D5F2 F3F0F4F0 40404040 = NA MN23040
000070 ==> 0075000F F5F5F0F0 F0F0F040 40F9F2F1 = ...5500000 921
000080 ==> F2F3F140 40404040 40D9E4D5 40C4C1E3 = 231 RUN DAT
000090 ==> C5404000 00610000 61000040 40404040 = E ../..
0000A0 ==> 4040C5D4 D7D3D6E8 C5C540C3 D6D4D7C5 = EMPLOYEE COMPE
0000B0 ==> D5E2C1E3 C9D6D540 D9C5D7D6 D9E34040 = NSATION REPORT
0000C0 ==> 40404040 40404040 40404040 40404040 =
0000D0 ==> D7C1C7C5 40000000 40404040 40404040 = PAGE ...
0000E0 ==> 40404040 40404040 40404040 40404040 =
0000F0 ==> 40404040 404040C8 C9D9C540 40E8D9E2 = HIRE YRS
000100 ==> 40D6C640 40404040 40404040 40404040 = OF
000110 ==> 40404040 40404040 40404040 40404040 =

```

Conditional breakpoints can be set via the WHEN command. To pause execution when the 9 bytes of data located at address 3789F3D0 is less than S05000000, enter WHEN 3789F3D0 I(9) < 'S05000000'. GO will resume program execution.


```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> when 3789f3d0 1(9) < 's05000000';go          SCROLL ==> CSR
PROGRAM: CWXTSUBC  MODULE: CWXTCOB  COMP DATE: 08/14/2006  COMP TIME:11:18:21

SYSTEM K R2 > 3789F3D0

SYSTEM K R6 > 3789F3D0
** END **

----- After CWXTSUBC:132/AMODE 31 ->
====> B 1862 LR R6,R2 *..*
000133 5820 1004 L R2,4(R1) *..*
000134 4120 2000 LA R2,0(R2) *..*
000135 5020 9138 ST R2,312(R9) *8.j.*
000136 1872 LR R7,R2 *..*
000137 5820 1008 L R2,8(R1) *..*
000138 4120 2000 LA R2,0(R2) *..*
000139 5020 913c ST R2,316(R9) *8.j.*
000140 D203 D088 A034 MVC 136(4,R13),52(R10) *K.)h.*
000141 8F2F 9158 ICM R2,15,344(R9) *..j.*
000142 5880 C028 L R11,40(R12) *..*
000143 4780 B124 BE/BZ 292(R11) *..*
000144 5830 905C L R3,92(R9) *..*
000145 58F0 30F4 L R15,244(R3) *.0.4*
000146 4110 A153 LA R1,339(R10) *..*

:02.5 02/50

```

Program execution is paused when the conditional expression is met.

```

SESSION1 - EXTRA Personal Client
----- XPEDITER/TSO - SOURCE -----
COMMAND ==> - WHEN 3789f3d0 1 (9) < 's05000000'          SCROLL ==> CSR

SYSTEM K R2 > 3789F3D9

SYSTEM K R6 > 3789E039
** END **

----- Before CWXTSUBC/AMODE 31 ->
***** TOP OF MODULE *****
====> CWXTSUBC CSECT
000003 47F0 F028 B 40(R15) *.00.*
000004 DC XL8'00C3C5C500000110' *.CEE...*
000005 DC XL8'0000001447F0F001' *....00.*
000006 DC XL4'98CEAC00' *q...*
000007 375BF79E DC A(CWXTSUBC+X'B6')
000008 DC 12X'00'
000009 90EC D00C STM R14,R12,12(R13) *.}.*
000010 4110 F038 LA R1,56(R15) *.0.*
000011 98EF F04C LM R14,R15,76(R15) *q.0<.*
000012 07FF BR R15 *..*
000013 DC XL2'0000' *..*
000014 375BF6E8 DC A(CWXTSUBC) *..*
000015 DC XL4'00000000' *...*

:00.1 02/15

```

As you can see, when you are using Xpediter's sourceless debugging support, you can still use many of the features you are familiar with today. Even though you may not have a source listing to test with, you can still take advantage of Xpediter to set breakpoints, intercept abends, step through instructions and modify data while testing your program.