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.



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 - EXTRAI Pe	rsonal Client				_ 0 X
		XPEDITER/T	SO - AT DIS	SPI AY	
COMMAND ===>		All Los I Ling I			SCROLL ===> CSR
	AT ADDRES	S 375FA6F8	: 47F0F028	в :в 40(
	AT ADDRES	S STIENCE			WXTSUBC/AMODE 31
BASE = 375EA6E8	0 - 2 -	4 - 6 -	8 - A -	C - E - =	0-2-4-6-8-A-C-E-
********	******	******* TOP		*********	***************
000000 ===>	00F0F028	00c3c5c5	00000110	00000014 =	.00CEE
	47F0F001	98CEAC00	375EA79E	00000000 =	
	00000000	00000000	90ECD00C	4110F038	
	98EFF04C	07FF0000	375EA6E8	00000000 =	
	375EADE0	375EA796	375EA6E8	0012FA38 =	
	375E9888	375EA782	00104001	00000008 =	
	C3E6E7E3	E2E4C2C3	F2F0F0F6	F0F8F1F4 =	
	F1F1F1F8	F2F1F0F3	F0F1F0F1	04740000 =	
	0000076C	A0687C4C	20000000		
			00000400		
	01000000	08000000		00000017 =	
	0000001B	0008000	40404040	0008C3E6 =	
0000B0 ===>		c2c30500	0001375E	9AD80000 =	
	0000FFFF	FFB2375E	A6E8375E	A7200000 =	
	00080000	0006375E	A720375E	9AD8375E =	
	9AB00000	00050000	00000000	00000000 =	
	00000000	00000000	00000000	0000001 =	
	40404040	40404040	40404040	40404040 =	:
000110 ===>	40404040	40404040	40404040	40400000 =	
2 <mark>11</mark>	@: 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 - EXTRAI Personal C	llient		- 0
	XPEDITER/1	SO - AT DISPLAY	
COMMAND ===> ps cwx	tsubc:		SCROLL ===> CSR
	ADDRESS 375EA6E8	: 47F0F028 : I	в 40(,15)
			fore 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 ===> 00F0	F028 00c3c5c5	00000110 000000	014 = .00CEE
000010 ===> 47F0	F001 98CEAC00	375EA79E 000000	
000020 ===> 0000	0000000 000000	90ECD00C 4110F0	038 =
000030 ===> 98EF	F04C 07FF0000	375EA6E8 000000	000 =0<;.Y
000040 ===> 375E	ADE0 375EA796	375EA6E8 0012FA	\38 = .;[\.;;.Y
000050 ===> 375E	9888 375EA782	00104001 000000	
000060 ===> C3E6	E7E3 E2E4C2C3	F2F0F0F6 F0F8F1	LF4 = CWXTSUBC20060814
000070 ===> F1F1	F1F8 F2F1F0F3	F0F1F0F1 047400	000 = 111821030101
000080 ===> 0000	076C A0687C4C	2000000 008001	LO9 =%@<
000090 ===> 0100	00000 08000000	00000400 000000	017 =
0000A0 ===> 0000	00018 00008000	40404040 0008c3	3E6 =
0000B0 ===> E7E3	E2E4 C2C30500	0001375E 9AD800	00 = XTSUBC;.Q
0000c0 ===> 0000	FFFF FFB2375E	A6E8375E A72000	000 =;.Y.;
0000D0 ===> 0008	0000 0006375E	A720375E 9AD837	
0000E0 ===> 9AB0	0000 00050000	000000 0000000	=
0000F0 ===> 0000	0000000 000000	000000 0000000	01 =
000100 ===> 4040	4040 40404040	40404040 404040	
000110 ===> 4040	4040 40404040	40404040 404000	= 000
<u>20</u>	@: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 - EXTRAI Personal Client	ŧ		X
	XPEDI	TER/TSO - SOURCE	
COMMAND ===> f r6 all			SCROLL ===> CSR
	DULE: CWX	COB COMP DATE: 08/14/2006	
** END **			
			XTSUBC/AMODE 31 ->
**************	******	TOP OF MODULE *************	**************
000002	CWXTSU	3C CSECT	
000003 47F0 F028	В	40(,R15)	*.00.*
000004	DC	xL8'00c3c5c500000110'	*.CEE*
000005	DC	XL8'0000001447F0F001'	*00.*
000006	DC	XL4'98CEAC00'	*q*
000007 3758F79E	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'	* *
Ø:	00.2		02/23

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 - EXTRAI Personal Client			_0×
	XPEDI	TER/TSO - SOURCE	
COMMAND ===> go_			SCROLL ===> CSR
	1	8 CHARS 'R6' FOUND	
** END **			
		Before	CWXTSUBC/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 132 1862	LR	R6,R2	**
000133 5820 1004	L	R2,4(,R1)	* *
000134 4120 2000	LA	R2,0(,R2)	k k
000135 5020 9138	ST	R2,312(,R9)	*8.1.*
000136 1872	LR	R7, R2	* #
000137 5820 1008		R2,8(,R1)	* *
000138 4120 2000	LA	R2,0(,R2)	* *
000139 5020 913c	ST	R2,316(,R9)	*8. 1 *
000140 D203 D088 A034	MVC	136(4,R13),52(R10)	*K.]h*
000141 BF2F 9158	ICM	R2,15,344(R9)	*j.*
Ø:00.	3		02/17

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 - EXTRAI	Personal Client			_ <i>a</i> x
		KPEDIT	ER/TSO - SOURCE	
COMMAND ===>				OLL ===> CSR
		EFORE	BREAKPOINT ENCOUNTERED	
**	END **			
			Before CWXTSUBC:13	2/AMODE 31 ->
	70 9138	L	R7,312(,R9)	"]."
	60 9134	L	R6,308(,R9)	*].*
	20 1000	L	R2,0(,R1)	**
000130 41	20 2000	LA	R2,0(,R2)	**
000131 50	20 9134	ST	R2,308(,R9)	*&.j.*
====> B 18	62	LR	R6,R2	**
000133 58	20 1004	L	R2,4(,R1)	**
000134 41	20 2000	LA	R2,0(,R2)	k k
000135 50	20 9138	ST	R2,312(,R9)	*8. j.*
	72	LR	R7,R2	* *
	20 1008	L	R2,8(,R1)	* *
	20 2000	ŪA –	R2,0(,R2)	* *
	20 913c	ST	R2,316(,R9)	*8.1.*
	03 D088 A034	MVC	136(4,R13),52(R10)	*K. 1h*
	2F 9158	ICM	R2,15,344(R9)	*j.*
20	@:00.6			02/25

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 - EXT							_ 0 X
COMMAND = PROGRAM:	≔=> go ha	If MODULE:					SCROLL ===> CSR COMP TIME:11:18:21
SYSTEM K	R2				>	3789F3D0	
SYSTEM K					>	0000000	
000127	** END ** 5870 913	e		7,312(,	 	Before CWXTSU	JBC:132/AMODE 31 ->
000128	5860 913		L R	16,308(,	R9)		*[2]]*
000129	5820 100			2,0(,R1			**
000130	4120 200			R2,0(,R2			**
000131	5020 913	4		2,308(,	R9)		*&.j.*
====> B	1862			R6,R2			**
000133	5820 100			2,4(,R1			**
000134	4120 200			2,0(,R2			* *
000135	5020 913			2,312(,	R9)		*&.j.*
000136	1872			17,R2			**
000137	5820 100			12,8(,R1			**
000138	4120 200			2,0(,R2			* * *
000139	5020 913			2,316(,			*&.j.*
000140	D203 D08			136(4,R1		(R10)	*K.}h*
000141	BF2F 915	8	ICM P	2,15,34	4(R9)		*j.*
ZL		@: 00.1					02/22

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			. 0 X
	XPEDI	TER/TSO - SOURCE	
COMMAND ===> memory 3789f3			CROLL ===> CSR
		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)	*].*
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		*&.j.*
000140 D203 D088 A034	MVC	136(4,R13),52(R10)	*К.}h*
000141 BF2F 9158	ICM	R2,15,344(R9)	*].*
Ø:02.4	1		02/30

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

SESSION1 - EXTRA! Personal Client				_0 X
	XPEDITER/T	SO - MEMORY		
COMMAND ===> locate *				SCROLL ===> CSR
PROGRAM: CWXTSUBC MODULE	E: CWXTCOB	COMP DATE:	08/14/2006	COMP TIME:11:18:21
				BC:132/AMODE 31
BASE = 3789F3D0 0 - 2 -	4 - 6 -	8 - A -		0-2-4-6-8-A-C-E-
*****	******* TOP	OF DATA **	***********	******
000000 ===> E2F0F5F5	FOFOFOFO	F0000000	0F000000 =	s05500000
000010 ===> 0108300F	0108300F	00000000	00000000 =	
000020 ===> 00010830	0F000000	000F0000	00000000 =	
000030 ===> 0000000F	00000000	FOF2F3F3	F4E2F1D9 =	02334s1R
000040 ===> D6E840C7	c1b9c4b5	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 =	
Ø:00.3	3			02/23

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 - EXT	RAI Personal C	lient				X
		XPE	DITER/1	rso - sourci	E	
COMMAND =	==> when 3	789f3d0 1((9) < 's	so5000000';	90	SCROLL ===> CSR
PROGRAM:	CWXTSUBC	MODULE: C	WXTCOB	COMP DATE	: 08/14/2005	COMP TIME:11:18:21
SYSTEM K	R2			>	3789F3D0	
of of other the						
SYSTEM K				>	3789F3D0	
	** END **					
8	1060			• 2	- After CWXTSU	UBC:132/AMODE 31 ->
====>> B 000133	1862 5820 1004	LR		4(,R1)		e e
000134	4120 2000			0(,R2)		
000135	5020 9138			312(,R9)		*8 1 *
000136	1872	LR		R2		***
000137	5820 1008			8(,R1)		* *
000138	4120 2000	LA LA		0(,R2)		* *
000139	5020 913c			316(,R9)		*8.j.*
000140	D203 D088	A034 MV	C 136	(4,R13),52	(R10)	*к.}h*
000141	BF2F 9158			15,344(R9)		*j.*
000142	58B0 C028			L,40(,R12)		*{.*
000143	4780 B124		/BZ 292			**
000144	5830 905c			92(,R9)		* **
000145	58F0 30F4			5,244(,R3)		*.0.4*
000146	4110 A153		RI,	339(,R10)		1
20		@:02.5				02/50

Program execution is paused when the conditional expression is met.

SESSION1 - EXTRA	Personal Client			_ <i>a</i> x
		XPEDIT	TER/TSO - SOURCE	
COMMAND ====	>			SCROLL ===> CSR
	-	WHEN 3789	9f3d0 1 (9) < 's05000000'	
SYSTEM K R	2		> 3789F3D9	
SYSTEM K R	END **		> 3789E039	
	END **		Poforo (CWXTSUBC/AMODE 31 ->
*******	*******	*******	TOP OF MODULE **********	WATSUBC/AMOUE 31 ->
====>		CWXTSUE		
000003 4	7F0 F028	8	40(,R15)	*.00.*
000004		DC	xL8'00C3C5C500000110'	*.CEE*
000005		DC	XL8'0000001447F0F001'	*00.*
000006		DC	XL4'98CEAC00'	*q*
000007 3	758F79E	DC	A(CWXTSUBC+X'B6')	
000008		DC	12x'00'	
	OEC DOOC	STM	R14,R12,12(R13)	*}.*
	110 F038	LA	R1,56(,R15)	*0.*
	8EF F04C	LM	R14,R15,76(R15)	*q.0<*
	7FF	BR	R15	×
000013	750 04 00	DC	XL2'0000'	×., ×
	758F6E8	DC	A(CWXTSUBC)	
000015		DC	xL4'0000000'	~····*
20	@:0	0.1		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.