



**XDC/SE™
RELEASE GUIDE**

XDC/SE™ Release S2.0 for OS/390

David B. Cole

XDC/SE™ S2.0 RELEASE GUIDE

PREFACE

PROPRIETARY LEGEND

XDC/SE™ and its documentation (collectively, "Product"), including copies thereof, are the confidential and proprietary property of Cole Software, LLC ("Owner"). The Product may be used only by those organizations that are licensed by Owner for such use and only in the manner so licensed. The program and documentation may not be published, reproduced, distributed, or made available to third parties for any purpose without the expressed written permission of Owner; however, a reasonable number of copies may be made of the documentation (including the copyright notices and proprietary legends thereon) as is necessary for the legitimate use of the Product within a licensed organization.

Except as may be otherwise expressed in a signed licensing agreement between Owner and Customer, Owner makes no representations or warranties, expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose, the warranty of freedom from rightful claims by way of infringement and the like, and any warranty as to accuracy.

WARNING! XDC/SE™ is a powerful tool for dynamically locating and correcting malfunctions in actively executing user programs and operating system programs and subroutines. Accordingly, it is inherent in its design, that unless the use of this Product is properly controlled, then under certain conditions a malicious or careless user can use the Product to alter, subvert, counterfeit, damage or otherwise disturb the normal execution of user programs or system routines including, under certain conditions, both its own and system security routines.

Therefore, even if advised of the possibility of loss or damages, under no circumstances shall Owner be liable for any loss or damage whatsoever (including death) arising from the Product, whether such loss or damage be direct, indirect, consequential, special or otherwise. Further, Owner shall not be obligated to indemnify any user of the Product in any manner for any loss which the user or anyone else may experience, of any kind or nature, arising out of the use or misuse of the Product.

CONTACTING COLE SOFTWARE

The **XDC/SE™** products are marketed by **COLE SOFTWARE, LLC** with its principal office in Afton, Virginia. If you would like more information, please contact COLE SOFTWARE Marketing as follows:

Phone: **800-XDC-5150** or **520-771-2003**
FAX: **520-771-2005**
E-Mail: **bshimizu@colesoft.com**
Home Page: **<http://www.colesoft.com>**

Our Technical Support contacts are:

Phone: **540-456-8210**
FAX: **540-456-6658**
E-Mail: **cswtech@colesoft.com**
Home Page: **<http://www.colesoft.com>**
FTP site: **<ftp.colesoft.com>**

XDC/SE™ S2.0 RELEASE GUIDE

(Preface)

Our Customer Services contacts are:

Phone: **540-456-8210**
FAX: **540-456-6658**
E-Mail: **cswstaff@colesoft.com**
Home Page: **http://www.colesoft.com**

Our snail mail address is:

Address: **Cole Software, LLC**
 736 Fox Hollow Road
 Afton, Virginia 22920
 USA

Our home page provides the following services:

- General information about XDC/SE.
- E-mail links to both Marketing, Technical Support, and Customer Services.
- FTP links for uploading diagnostic information and other files to Technical Support.
- FTP links for downloading current maintenance for XDC/SE.
- Links permitting customers to download a full set of XDC/SE's documentation in any of several document formats.

TRADEMARKS

XDC/SE™, TFS™, XDC-TFS™, CDF™, XDC-CDF™, and FASM™ are trademarks of Cole Software, LLC. **Extended Debugging Controller®** and **XDC®** are registered trademarks of Cole Software, LLC. Other brand and product names referenced in this document are trademarks or registered trademarks of their various holders. Use of their names herein is for identification purposes only.

ABOUT THIS MANUAL

This manual contains an extraction of panels from XDC/SE's Online Help database. These panels describe in detail how to use XDC/SE.

This manual was created by running XDC/SE with a special User Communications Interface exit routine[†] named XDC-ONLHG. XDCONLHG sends commands to XDC/SE that causes it to generate all of its Online Help frame displays. XDC-ONLHG then receives the generated frames from XDC/SE and saves them for processing later by a word processing program. Note, the source code for XDCONLHG can be found in the SRCONLHG member of XDC/SE's XDCASM library.

On the following pages each header shows the XDC/SE command that generated the information that follows. Note that the keywords used in the various HELP commands can almost always be abbreviated. The minimum abbreviations are shown both in the table of contents and in the section headers via uppercase letters. Caution: The abbreviations indicated in this way are abbreviations for the names of Online Help panels, not for the commands that they may describe.

[†]See HELP USERCOMM UCI.

XDC/SE™ S2.0 RELEASE GUIDE

(Preface)

ADDITIONAL MANUALS

XDC/SE customers may make as many copies of this manual as they feel is necessary for the legitimate use of XDC/SE within their organization. For copying purposes, if this manual is bound together by plastic Velo-Bind straps, then it may be easily and safely debound simply by running a hacksaw blade between the binding strap and the back cover.

Customers may download from our web site (www.colesoft.com) printable copies of all of XDC/SE's manuals. Each manual is available in both PDF and HTML formats.

In addition, all manuals can be printed directly from within XDC/SE itself. To print your own set of manuals, start an XDC/SE debugging session (example: XDCCALL IEFBR14), then issue the following commands:

```
PRINT HELP USERGUIDE;SET PRINT CLOSE  
PRINT HELP COMMANDS;SET PRINT CLOSE  
PRINT HELP MESSAGES;SET PRINT CLOSE  
PRINT HELP WHATSNEW S20;SET PRINT CLOSE
```

Alternatively, you also can print these manuals by issuing XDC/SE's **READ** command against the **MANUALS** member of XDC/SE's command file library: **READ dsname(MANUALS)**. Unfortunately, I cannot tell you what the library's dsname is except that its lowest level qualifier probably is **XDCCMDS**. That's because the library's actual dsname is dependent upon decisions made by the System's Programmer who installed XDC/SE at your data center, so you will have to ask him for this information.

You also may print a **Quick Reference** for XDC/SE by issuing XDC/SE's **READ** command against the **QUICKREF** member of XDC/SE's command file library: **READ dsname(QUICKREF)**.

For more information about using the **PRINT HELP** and related commands, see **HELP HELP PRINTING**.

XDC/SE™ S2.0 RELEASE GUIDE

CONTENTS

PREFACE	ii
PROPRIETARY LEGEND	ii
CONTACTING COLE SOFTWARE	ii
TRADEMARKS	iii
ABOUT THIS MANUAL	iii
ADDITIONAL MANUALS	iv
CONTENTS	v
INTRODUCTION	1
A Roadmap	1
Online Help Panels	3
Help Whatsnew	3
Help Whatsnew S20M	3
Help Whatsnew S20	4
Help Whatsnew S20 Breakpoints	5
Help Whatsnew S20 Breakpoints Conditional	5
Help Whatsnew S20 Breakpoints Disable	6
Help Whatsnew S20 Breakpoints Storealter	6
Help Whatsnew S20 Breakpoints Watch	7
Help Whatsnew S20 Breakpoints X	8
Help Whatsnew S20 Commands	8
Help Whatsnew S20 Commands Linecmds	12
Help Whatsnew S20 Commands Deleted	12
Help Whatsnew S20 Commands Cmdlibrary	13
Help Whatsnew S20 Hookprocessing	13
Help Whatsnew S20 Languageenvironment	14
Help Whatsnew S20 SScrolling	14
Help Whatsnew S20 SOurcelevel	15
Help Whatsnew S20 Miscellaneous	16
INDEX	17

XDC/SE™ S2.0 RELEASE GUIDE

XDC/SE™ S2.0 RELEASE GUIDE

INTRODUCTION

Cole Software has pursued the goal of making XDC/SE's online documentation as comprehensive as possible. Towards that end, we have devoted considerable effort to greatly expanding the amount of information online and to improving the accessibility of that information and the navigability of the Online Help database as a whole.

This manual is nothing more than a printout of a section of the Online Help database. It is provided for those people (like myself) who steadfastly prefer looking at paper instead of glass. (It's hard to write margin notes on glass.)

The information in the Online Help database has been segmented into five printed documents:

- **XDC/SE™ User Guide**
Contains comprehensive tutorials about the many features and capabilities of XDC/SE.
- **XDC/SE™ Commands**
Contains the detailed syntax, usage descriptions, and examples of all of XDC/SE's commands.
- **XDC/SE™ Messages**
Contains descriptions of all of the messages that can be issued by XDC/SE and all of its various components.
- **XDC/SE™ S2.0 Release Guide**
Contains a history of all changes and upgrades made in the current release of XDC/SE.
- **XDC/SE™ Quick Reference**
Contains brief lists of XDC/SE commands, built-in equates, and other information.

There are a couple of important structural differences between the Online Help and these manuals:

- When the Help panels are displayed online, a large number of "hyperlinks" are available for easily pursuing subjects related to the current information. These hyperlinks do not exist in the printed manuals.
- The printed manuals contain comprehensive indexes to help you quickly find the specific information that you may be looking for. These indexes do not exist online.
- "Release Guides" for older versions and releases of XDC/SE are available online via the "HELP WHATSNEW" command.

A Roadmap

The structure of this manual follows the structure of the Online Help database. A consequence of this is that the sequence of information in this book, over all, is decidedly non-linear. For those of you who prefer to read a manual from beginning to end, please accept my apologies. However, please let me make some suggestions.

If you are an experienced XDC/SE user, then start with the **XDC/SE™ S2.0 Release Guide**. This will tell you what's new in this release of XDC/SE. Online, the Release Guide can be reached by typing HELP WHATSNEW. You can then use hyperlinks to pursue the specific information that is of interest to you.

XDC/SE™ S2.0 RELEASE GUIDE

(Introduction)

For new users, turn to the **XDC/SE™ User Guide**, and examine its Table of Contents carefully. You will see that there are about two dozen major topics arranged alphabetically: Addressing, Attentions, Breakpoints, ..., Virtmem, XDCCALL. Information within topics is presented more or less linearly. The following **User Guide** topics are of particular interest:

- Perhaps the first topic that should be read is named "**DEBUGGING**". This and its subtopics give comprehensive information about whether and to what extent you may have to modify your program in order to use XDC/SE.
- Another topic that should be read early on is named "**XDCCALL**". XDCCALL is a utility program that can be used to start a debugging session for your program.
- If you plan to debug programs that run as batch jobs or system tasks, then read the "**CDF**" topic. "Cross Domain Facility" is the component of XDC/SE that permits user terminals to connect to debugging sessions for background jobs.

For XDC/SE command information, turn to the **XDC/SE™ Commands** manual. Start with the basic commands. The DISPLAY, FORMAT, and LIST commands display storage and important program related structures. The AT and TRAP commands set breakpoints. You can use the TRACE command to step execution through your program slowly. The ZAP command allows you to change storage and registers.

If you wish to play with XDC/SE's terminal and user interfaces, read the "**FULLSCREEN**" section of the **User Guide**. Also, try the PROFILE command for displaying and changing a very large number of session parameters.

Generally, the best approach is to plan your reading using the Table of Contents. And of course, if you can't find the information that you are looking for, call us. There's no charge, and we will be glad to help! Our number is 540-456-8210. If the information that you want is in the book, we will explain what you want to know and tell you where to find complete information. If it is not, then we will add it for our next release.

XDC/SE™ S2.0 RELEASE GUIDE

Online Help Panels

Help Whatsnew

XDC/SE's Change History: For detailed information, type S at the left, then press ENTER. The information presented will be the most useful to experienced XDC/SE users who want a concise summary of what has changed and a road map of where to look for more specific information.

- S2.0 - (10/00) Incremental changes implemented via maintenance.
- S2.0 - (08/00) New release: Source Level Debugging Support!
- S1.0 - (11/98) New version! PDS/E support! XMS Support! Etc.
- X3.3 - (10/97) Incremental fixes and additions
- X3.2 - (12/96) Incremental fixes and additions
- X3.1 - (04/95) Beta-test release of X3.2
- X3.0 - (06/94) MVS/ESA support

Help Whatsnew S20M

The following changes have been implemented via maintenance to release S2.0 of XDC/SE. The applicable maintenance level is S20-0011D. (Maintenance may be obtained from Cole Software's web site. The URL is <http://www.colesoft.com>.)

- The description of line commands has been moved from HELP FULLSCREEN LINECMDS to **HELP LINECMDS**.
- In the printed documentation, the descriptions of both line commands and command library files have been moved from the User's Guide to the Commands Reference.
- Documentation has been added to the Online Help for the following commands:
 - SET ASIS
 - SET UPCASE
 - LIST UPCASE
- The following significant topics have been added to the Online Help:
 - HELP ADDRESSING SYMBOLIC MIXEDCASING**
Describes mixed case support as it applies to symbolic names.
 - HELP COMMANDS LIST UPCASE**
Also provides in depth information about mixed case support.
 - HELP DEBUGGING TIPS 0C4**
Discusses the reasons why using XDC/SE can sometimes appear to cause 0C4s (and other abends) in user code.
 - HELP MULTITASK SCOPE**
Discusses the scope of a debugging session and what happens when XDC/SE receives control under a task that is not within the proper scope.
- The influence of the SET ASIS command has been expanded to affect the following:
 - **Zapping:** Lower case letters can now be zapped into the text portion of hex-text displays. See HELP LINECMDS Z for more information.
 - **CSECT Names:** Mixed casing can now be using for csect names, entry point names, and other ESD symbols. This is because the Binder supports mixed case

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20M)

names, and the High Level Assembler's ALIAS statement can be used to map Assembler-known external symbols (which the Assembler internally manages as uppercase names) to mixed case Binder-known names (which the Binder processes with case sensitivity).

- **DSECT Names:** Mixed casing can also be used for dsect names, since the DMAP command supports the reading of csect maps as if they were dsect maps.
- **Other Names:** Mixed casing is **NOT** supported for other kinds of names. Specifically, it is not supported for:
 - Load module names
 - Program object names
 - Machine instruction names
 - Field names
 - Equate names
 - Breakpoint names
 - Hook namesAll references to these names will always be upcased regardless of whether or not SET ASIS is in effect.
- **SCANLOG Command:** When case sensitivity is in effect (i.e. the SET ASIS command has been issued), the session log search performed by the SCANLOG command either will or won't be case sensitive depending upon whether the string being search for is or is not enclosed within quotes (').

For more information, please see HELP ADDRESSING SYMBOLIC MIXEDCASING.

- The commands listed below all accept text strings given either enclosed or not enclosed within quotes ('). The text strings will or won't be accepted as mixed case strings as follows:
 - Unquoted strings will always be accepted as mixed case strings regardless of whether or not case sensitivity is in effect.
 - Quoted strings either will be accepted as mixed case strings or will be upcased according to whether or not case sensitivity is in effect.

The commands in question are:

- ALARM
- COMMENTARY
- SET KEYS nn
- LIST NOTES
- NOTES
- TSO

Help Whatsnew S20

XDC/SE S2.0 contains significant new capabilities, chief among them being source level debugging support. XDC/SE now has the ability to read ADATA records produced by IBM's High Level Assembler and construct from them comprehensive maps of a program's code in storage. For specific information about this and other new features, select the following topics.

- **BREAKPOINTS:** The capabilities and features of XDC/SE's breakpoints have been enhanced in several ways. For more information, see HELP WHATSNEW S20 BREAKPOINTS.
- **COMMANDS:** XDC/SE supports several new commands. Also, the processing and capabilities of some commands have been changed, and a few commands have been deleted. For more information, see HELP WHATSNEW S20 COMMANDS.
- **HOOKPROCESSING:** Hook processing has been improved. See HELP WHATSNEW S20 HOOKPROCESSING for more information.

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20)

- **LANGUAGEENVIRONMENT:** XDC/SE now interfaces with Language Environment routines to determine who should handle a particular abend, L.E. or XDC/SE. For more information, see HELP WHATSNEW S20 LANGUAGEENVIRONMENT.
- **LONGNAMES:** Full support is finally available in XDC/SE for displaying and using long names (names longer than 8 characters). For more information, see HELP WHATSNEW S20 SOURCELEVEL.
- **PF-KEYS:** Factory default PF-key settings have been changed to accommodate the new LEFT/RIGHT scrolling commands. For more information, see HELP WHATSNEW S20 SCROLLING.
- **SCROLLING:** XDC/SE now supports left/right scrolling. For more information, see HELP WHATSNEW S20 SCROLLING.
- **SOURCELEVEL:** XDC/SE can build comprehensive source program maps from ADATA produced by IBM's High Level Assembler. These maps include images of the machine instructions as originally coded by the programmer as well as program commentary and various other source level structures. For more information, see HELP WHATSNEW S20 SOURCELEVEL.
- **ZAPPING:** XDC/SE can now zap into store protected storage when running authorized. For more information, read about the ZAP command in HELP WHATSNEW S20 COMMANDS.
- **MISCELLANEOUS:** For information about various additional changes to XDC/SE, see HELP WHATSNEW S20 MISCELLANEOUS.

Help Whatsnew S20 Breakpoints

The capabilities and features of XDC/SE's breakpoints have been enhanced in several ways. For specific information, select the following topics. These topics can be selected directly, or use HELP *NEXT to proceed sequentially. Use HELP *FORWARD to skip.

- CONDITIONAL - Some new constants are now recognized in conditional expressions.
- DISABLE - Breakpoints can now be disabled without being purged.
- STOREALTER - Storage alteration tracing allows stopping execution either before or after an instruction that alters storage.
- WATCH - A "WATCH" is a kind of trap that can be used to monitor your program until a specified condition is met.
- X - The X line command can be used to toggle a breakpoint on and off.

Help Whatsnew S20 Breakpoints Conditional

Conditional expressions now accept two new boolean constants: TRUE and FALSE:

FALSE Example: **T B (FALSE)**

This is a conditional expression that always evaluates FALSE. A breakpoint having this expression will cause XDC/SE to receive control whenever the breakpoint is reached by user program execution, but unless other decisions occur, XDC/SE will always then allow user program execution to resume.

The (FALSE) expression is best used on breakpoints (traps or traces) that are used in conjunction with "WATCHs". (See HELP WHATSNEW S20 BREAKPOINTS WATCH.)

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Breakpoints Conditional)

Since WATCHs do not cause breakpoints to be set, a trace or a set of traps are needed in order for XDC/SE to have the opportunity from time to time to evaluate the WATCHs. The (FALSE) expression can be used with such a trace or set of traps to create the breakpoints necessary to make WATCHs useful without having those breakpoints participate in the stop/resume decisions being made by the WATCHs.

TRUE Example: **AT .MYSUB (TRUE)**

This is a conditional expression that always evaluates TRUE. It is, in effect, the default conditional expression. Using it is equivalent to omitting the conditional expression entirely. Support for the TRUE constant was written primarily for reasons of symmetry.

For more information, see **HELP BREAKPOINTS CONDITIONAL**.

Help Whatsnew S20 Breakpoints Disable

Breakpoints can now be disabled without being purged. In other words, the breakpoint's "opcode" is removed from user code, but its definition is not deleted. A disabled breakpoint definition is ignored during XDC/SE breakpoint analysis.

When all breakpoints for a particular location are disabled, the machine instruction's opcode at that location is restored. Enabling a breakpoint causes it to become effective again: The machine instruction's opcode is changed back to the breakpoint's opcode.

Being able to disable a breakpoint (instead of having to purge it) is very useful for managing complex breakpoint definitions such as those having conditional expressions and associated commands.

Help Whatsnew S20 Breakpoints Storealter

XDC/SE now has a storage alteration tracing capability that is very similar to its branch tracing facility. Whereas branch tracing causes XDC/SE to receive control at all branch-type instructions, the storage alteration trace causes XDC/SE to receive control at all store-type instructions. Conditional expressions can then be used to evaluate whether a particular storage alteration has occurred.

XDC/SE supports two kinds of storage alterations:

- A "stop before" trace stops on store-type instructions before they execute.
- A "stop after" trace stops on store-type instructions after they execute.

The advantage of a "stop before" trace is that execution is stopped prior to the expected damage occurring. On the other hand, the advantage of a "stop after" trace is that the trace's conditional expression can examine the alteration to determine whether or not it is the damaging one being sought for. This is particularly useful when the storage alteration problem being searched for involves the occasional storing of incorrect data into a field that is the target of a large number of stores of correct data.

A Cautionary Note: Although this "storage alteration" trace capability may sound exciting, in truth it is not really much more useful than a successful-branch-trace defined with a suitable conditional expression. The advantage of the storage-alteration trace is that it can detect damage exactly at the instruction that is inflicting the damage, while a branch-trace won't detect such damage until usually a few instructions later. The disadvantage is that this rather small improvement in precision comes at the cost of a substantial increase in processing overhead.

XDC/SE™ S2.0 RELEASE GUIDE

Help Whatsnew S20 Breakpoints Watch

A "WATCH" essentially is a conditional expression that is not associated with any particular breakpoint. Instead, whenever XDC/SE receives control for the purpose of evaluating a conditional event, all enabled WATCHs also are evaluated. If any WATCH, condition, or conditional expression evaluates TRUE, then XDC/SE stops the user program and allows the user to proceed with his debugging efforts.

Any number of WATCHs may be defined and enabled. In order to be useful, a WATCH should have a conditional expression assigned to it. A WATCH may also have one or more automatic commands associated with it as well. Different WATCHs may have different conditional expressions and automatic commands assigned. Whenever a particular WATCH evaluates TRUE, its associated commands (if any) will be executed. It is possible for multiple WATCHs to evaluate TRUE at the same time.

A WATCH by itself cannot cause XDC/SE to receive control. In order for a WATCH to work, XDC/SE must receive control for other reasons. But once XDC/SE has received control, then the WATCH's conditional expression is evaluated. If the result is TRUE, then XDC/SE will stop user program execution, execute any automatic commands that may be associated with the WATCH, and probably (depending upon the commands) present a display to the user.

On the other hand, if the condition's result is FALSE (and if all other conditions are FALSE), then XDC/SE will return control to the user program so that it can continue executing.

A WATCH in and of itself cannot cause XDC/SE to receive control. So in order for a WATCH to be effective, a program's execution has to be breakpointed from time to time by other means. Typically, such "other means" would be one or more conditional traces or traps.

Such other conditional traces or traps may themselves have any legal conditional expression, and each such expression will participate (along with the WATCHs) in XDC/SE's stop/resume decision process. So if particular trap points require unique considerations, then this is a useful facility.

On the other hand, when a trap point does not require a test condition different from the WATCHs, then you will have to provide a condition that always resolves to FALSE. There are of course many ways to construct such a condition ("`(10?,EQ,00000000)`" comes leaping to mind), but the easiest and best performing expression would be to use the newly supported boolean constant, FALSE. Example: **T B (FALSE)**. See HELP WHATSNEW S20 BREAKPOINTS CONDITIONAL for more information.

The usefulness of a WATCH arises from the fact that its conditional expression is evaluated each and every time XDC/SE receives control for the purpose of evaluating a conditional event. **Normally** when XDC/SE receives control due to a conditional trap or trace, only the conditional expression that is associated with that trap or trace is evaluated. **Normally** when XDC/SE receives control due to a conditional event (such as reaching a branch-type instruction during a T BY trace), only the event is evaluated. **But** when one or more WATCHs exist, their expressions also are evaluated, and XDC/SE will auto-resume the user program only if ALL of the conditions evaluate to FALSE! If any of the conditional expressions or events evaluate to TRUE, then XDC/SE will stop user program execution, present a display to the user, and let him issue commands.

EXAMPLE: Suppose that you have three conditions to test, any one of which will stop execution. Suppose that you want these conditions all tested at each of seven locations in your program. Without WATCHs, you would have to set three conditional traps (each one specifying one of the conditional expressions) at **each** of the seven locations (total: twenty-one traps).

Using WATCHs, you can simply define three WATCHs, each specifying one of the trapping conditions. Then you can just create seven conditional traps each using the special conditional expression, "`(FALSE)`". Thus, the seven traps serve as the

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Breakpoints Watch)

mechanism by which XDC/SE receives control, but they do not contribute to XDC/SE's decision as to whether or not user execution will be stopped. The following is a sample of the commands that might be used in this example:

Setting three WATCHs:

```
T W (R1,EQ,00000000):L REGS:WHERE
T W (R4?,NE,'I-CATCHR'):D R4?
T W (.DCBOFLGS,AND,10,EQ,00)
```

Setting seven non-stopping traps:

```
T .LOOP1 .ERR .ZPGM (FALSE)
T .LP5 .XYL .BYEARF .LASTHIT (FALSE)
```

WATCHs are managed similarly to traps:

- WATCHs are created by the TRACE command, displayed by the LIST BREAKPOINTS command, and disabled, reenabled, or purged by the SET BREAKPOINT command (just like ordinary breakpoints).
- When a WATCH is accepted, it is automatically disabled.
- WATCHs have family ID numbers just like other breakpoints. You can group multiple WATCHs into family sets and enable/disable them as needed.

For more information, see HELP BREAKPOINTS CONDITIONAL WATCH.

Help Whatsnew S20 Breakpoints X

"X" is a new line command that can be used to toggle a breakpoint on and off. For more information, see HELP LINECMDS X.

Help Whatsnew S20 Commands

The following commands have been added, changed or deleted for XDC/SE release S2.0:

AT

ATX

The AT, ATX, TRAP, and TRACE commands can be used to set breakpoints into and trace execution through program code that is located in store protected storage.

COPY

The copy command has been enhanced to accept register names as source or target locations. In addition, the source and target lengths can now be given as complex arithmetic expressions, not just simple constants. For more information, see HELP COMMANDS COPY.

DELETE CACHE

See SET CACHE command below.

DELETE MAPLIBS

See SET MAPLIBS command below.

DISPLAY

The displays produced by the DISPLAY command have been changed as follows:

- The fetch protection indicator (shown on each line just to the right of the display address) has been changed from "p" to "f".
- A store protection indicator has been added to the displays. It is an "s" shown along with the fetch protection indicator (when applicable).

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Commands)

The maximum permitted linecount value for the DISPLAY command has been increased from 500 to 1000.

DMAP

See MAP command below.

END

When a storage dump is produced as a result of the END DUMP command, the dump is no longer suppressible by OS/390's D.A.E. facility. (This is accomplished by setting the VRANODAE flag in the SDWA's Variable Recording Area.) This has been done because the user, when he issues the END DUMP command, has presumedly decided that he really truly does want the dump. For more information, see HELP COMMANDS END.

FIND

Code has been added to the FIND command to recognize and discard spurious hits that are nothing more than artifacts of FIND's own processing. For example, when the user types a FIND command, copies of that command wind up being created in several buffers of one sort or another, and these copies can wind up matching FIND's search. Now, however, code has been added to FIND's processing to attempt to recognize and discard these spurious hits.

FORMAT

The displays produced by the FORMAT command have been changed as follows:

- The FORMAT command can now produce source image displays of program code and control blocks when they are mapped by source image maps. For more information about source image maps, see HELP WHATSNEW S20 SOURCELEVEL.
- When a source image map applies to a display, whether or not it will be used for that display depends upon the setting of the SET FORMAT command. Choices are SOURCE, OBJECT, or BOTH. See HELP COMMANDS SET FORMAT for more information. In addition, SOURCE, OBJECT, and BOTH can be used as operands on the FORMAT command itself.
- When the FORMAT command displays a field that is exactly four bytes wide, it attempts to display an interpretation of that field as a pointer. (Previously, it attempted to display such fields as decimal numbers.)
- WHEN: The FORMAT command or DISPLAY command displays storage, AND: That storage is mapped by a Binder map, THEN: location offsets shown down the lefthand side of the display will be relative to the csect boundaries found within the Binder map. This is now true REGARDLESS of whether the Binder map was built by the MAP command or the DMAP command. (Previously, Binder maps that were built by the DMAP command would show offsets relative to the start of the map, not to the start of csects within the map.)
- When a dsect map overlays a csect map, then offsets for displayed locations within the dsect map will remain relative to the csect. (Previously, such offsets were shown relative to the dsect.)
- The maximum permitted linecount value for the FORMAT command has been increased from 500 to 1000.
- When conditional branch instructions are displayed, the FORMAT command will show the branch mnemonics where possible (e.g. BE BHR JLR etc.). These branch mnemonics are now reactive to their context. For example, if a "BC 8,---" instruction follows a LTR, it will display as a "BZ". On the other hand, if it follows a CR, it will display as a "BE".
- The display of machine instruction displacement fields has been changed. Now when those fields are displayed in hex, they are (a) always displayed as 3 digits long and (b) displayed within X-quote framing. Example:
LA R14,X'02C'(:,R14)

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Commands)

HDEFERRED

This is a new command that permits users to define deferred HOOKs in much the same way that XDC/SE supports deferred breakpoints. It permits the setting of HOOKs into load modules that have not yet been loaded into storage. For more information, see HELP COMMANDS HDEFERRED.

LEFT

RIGHT

XDC/SE now supports full implementations of the LEFT and RIGHT scrolling commands. In the factory default PF-key settings, LEFT and RIGHT are assigned to the 10th and 11th keys of key SET-B. (The corresponding keys in key SET-A are used for tracing commands.) For more information, see HELP COMMANDS LEFT and HELP COMMANDS RIGHT.

LIST CACHE

DELETE CACHE

When a MAP or DMAP command reads ADATA records from a MAPLIBS file, a filtered copy of that file is cached into storage for use in case the file needs to be read again in the future. The LIST CACHE command can be used to display information about the cached files, and the DELETE CACHE command can be used to force the purging of particular cache files. For more information, see HELP COMMANDS LIST CACHE and HELP COMMANDS DELETE CACHE.

LIST FEATURES

This is a new command that displays a list of special features installed on the current hardware and in the current OS/390 within which XDC/SE is running. For more information, see HELP COMMANDS LIST FEATURES.

LIST MAPLIBS

See SET MAPLIBS command below.

LIST PSW

LIST EPSW

These commands now accept a new keyword operand (FORMAT) to show an extended formatting of the PSW. For more information, see HELP COMMANDS LIST PSW and HELP COMMANDS LIST EPSW.

LIST REGS

LIST AREGS

These commands now highlight those registers whose contents have changed during tracing. For more information, see HELP COMMANDS LIST REGS and HELP COMMANDS LIST AREGS.

LIST Rn

LIST ARn

The displays produced by these commands can now be zapped. They also are point-and-shootable. For more information, see HELP COMMANDS LIST Rn and HELP COMMANDS LIST ARn.

MAP

DMAP

These commands now have the ability to read source level mapping information from ADATA records. (MAP and DMAP continue to be able to read SYM records as well.) Csect and dsect maps produced from ADATA records permit XDC/SE to display:

- Machine instructions exactly as coded by the programmer,
- Commentary written by the programmer,
- Other source program images when displaying code and control blocks in storage.

For more information, see HELP WHATSNEW S20 SOURCELEVEL.

ADATA records can be read from any place that they can found. This includes:

- From within program objects loaded from PDSE libraries.
- From within XOBJECT files located in sequential files or PDS/PDSE libraries.
- From SYSADATA files located in sequential files or PDS/PDSE libraries.

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Commands)

For more information, see HELP COMMANDS MAP and HELP COMMANDS DMAP.

REFRESH

This command causes XDC/SE's fullscreen management routines to fully repaint the display screen. The REFRESH command can be used in command files that may cause display disrupting messages to be issued outside of XDC/SE's awareness and control. The REFRESH command repairs a corrupted display by causing XDC/SE to send a complete display image to the workstation. (Normally, XDC/SE tries to keep track of what is being displayed and to send only modifications to the display.) For more information, see HELP COMMANDS REFRESH.

Generally this command is most useful when it is issued from an automatic source such as from an automatic command associated with a breakpoint or from a commands file being executed by the READ command. This is because when a workstation display is corrupted, it usually is not possible to issue commands from it. (Instead, press PA2.)

RIGHT

See LEFT command above.

SET BREAKPOINT

This is a new command that can be used to enable, disable, and purge breakpoints. It has greater functionality than the OFF command, and it is intended eventually to replace the OFF command. For more information, see HELP COMMANDS SET BREAKPOINT.

SET FORMAT

This is an old command that now has a new usefulness. Its SOURCE, OBJECT, and BOTH operands can be used to control whether or not source image maps are used by the FORMAT command when displaying storage to which such maps apply. For more information, see HELP COMMANDS SET FORMAT.

SET MAPLIBS

LIST MAPLIBS

DELETE MAPLIBS

These are new commands for managing lists of datasets and libraries containing ADATA records from which the MAP and DMAP commands can build source image maps of programs and control blocks in storage. When the MAP or DMAP command needs to read mapping information for a csect or dsect, it first searches the MAPLIBS list (if any) before looking within the load module or program object itself.

A MAPLIBS list can contain the names of both sequential and partitioned datasets. The SET MAPLIBS command is used to build the list, the LIST MAPLIBS command is used to display the list, and the DELETE MAPLIBS command is used to removed names from the list.

The SET MAPLIBS command also can be used to save one or more MAPLIBS lists into your debugging session's profile. Each saved list is assigned a name that later can be used to bring the list back into active usage. The LIST MAPLIBS and DELETE MAPLIBS commands can be used to display and purge the saved lists.

For more information, see HELP COMMANDS SET MAPLIBS, HELP COMMANDS LIST MAPLIBS, and HELP COMMANDS DELETE MAPLIBS.

SET WINDOW

This command has been changed so that it can set defaults for the LEFT/RIGHT scrolling commands:

- The SET WINDOW SCROLL ... command has been changed to SET WINDOW VERTICAL (The SCROLL operand remains supported for the time being.)
- A new HORIZONTAL operand has been implemented to support the setting of the default scrolling distance for the LEFT and RIGHT scrolling commands.

For more information, see HELP COMMANDS SET WINDOW.

SHOW

If:

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Commands)

- The SHOW command is used to display storage,
- And if the address expression used to reference that storage starts with the name of a symbol,
- And if that symbol is assigned to the storage location being displayed,
- And if that symbol has a length attribute other than zero,

THEN:

- The SHOW command will display the number of bytes (up to 16 bytes) indicated by the symbol's length attribute. (Whew!)

For more information, see HELP COMMANDS SHOW.

EXAMPLE: **SHOW .DCBDDNAM** displays 8 bytes of storage even though other symbols occur within the 8 bytes. (Previously, only 2 bytes of storage would have been shown because the DCBMACF1 symbol occurs at DCBDDNAM+2.)

HINT: If the SHOW command is displaying a number of bytes other than what you want, then use the EQUATE command to assign a symbol to the target location, and assign to that equate a suitable length attribute. Then use that equate's name on the SHOW command.

TRACE

TRAP

The TRACE, TRAP, AT, and ATX commands can be used to set breakpoints into and trace execution through program code that is located in store protected storage.

ZAP

The ZAP command can now be used to alter store protected storage.

With suitable authorization and security permits, XDC/SE can now zap store protected storage located even in other address spaces.

When zapping string data into a target, the ZAP command now supports a way to exclusive-OR the data into the target. For more information, see HELP COMMANDS ZAP BOOLEAN.

For a description of new and changed line commands, see HELP WHATSNEW S20 COMMANDS LINECMDS.

For a list of deleted commands, see HELP WHATSNEW S20 COMMANDS DELETED.

For a description of new and changed command library files, see HELP WHATSNEW S20 COMMANDS CMDLIBRARY.

Help Whatsnew S20 Commands Linecmds

The following line commands have been added or changed for XDC/SE release S2.0:

K

This command sets a hook at the location being displayed. For more information, see HELP LINECMDS K.

X

This command toggles a breakpoint on and off. For more information, see HELP LINECMDS X.

Help Whatsnew S20 Commands Deleted

The following commands are no longer recognized by XDC/SE:

DELETE SLISTLIBS
LIST SLISTLIBS

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Commands Deleted)

SET SLISTLIBS
SMAP
TRACE S (source statement tracing)

Help Whatsnew S20 Commands Cmdlibrary

The following command library files have been added, changed, or deleted for XDC/SE release S2.0:

EREGPTRS

This file can be used to assign a set of floating equates to label the locations, within the error level Primary Address Space, to which the error level general registers may point. For more information, see HELP CMDLIBRARY EREGPTRS.

FASMREGS

This file can be used to assign a set of floating equates to label the locations within a dataspace to which the general registers may point. For more information, see HELP CMDLIBRARY FASMREGS.

HOOK

This command file has been deleted. Its functionality has long since been replaced by XDC/SE's HOOK command. (If you really really hate to see this command file go, then just save a copy from XDC/SE's prior release into your own private library.) See HELP COMMANDS HOOK for more information.

HOOKPLI

This command file has been deleted.

ISPF410 (deleted)

ISPF430 (deleted)

ISPF4XX (added)

The various ISPF4rm command files have been replaced by ISPF4XX because the same sequence of XDC/SE commands work correctly for disabling ISPF's internal error recovery logic for all releases of ISPF from before release 4 through at least 4.8.

QUICKREF

This file has been changed: A command line has been added to include information about point-and-shoot commands. For more information, see HELP CMDLIBRARY QUICKREF.

REGPTRS

This file has been changed: The creation of @ERn equates (for labeling the locations pointed to by the error level general registers) has been split out of this file and moved to the new EREGPTRS file. See HELP CMDLIBRARY REGPTRS for more information.

Help Whatsnew S20 Hookprocessing

The following improvements have been made to the processing of hooks that have been reached by user program execution:

- The HOOK SVC routine no longer uses a #DIE macro to start the debugging session. Instead, it uses a method that starts the session with **both** the error level environment and retry level environment set to the user program's environment. (Previously, the error level environment was the hook SVC routines environment. This tended to be confusing.)
- Error messaging has been significantly improved:

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 Hookprocessing)

- When a hook in common storage has to be bypassed because of being executed from the "wrong" address space, the DBC964W messages now display the ASIDs for both the intended and actual address spaces.
- When an attempt to remove a hook from storage fails, the DBC965T messages now provide comprehensive information regarding the reason why the failure occurred.

For more information about using hooks, see HELP DEBUGGING HOOKS.

Help Whatsnew S20 Languageenvironment

The usage of XDC/SE with programs running within IBM's Language Environment (L.E.) has been improved. XDC/SE now interfaces with an LE service routine (CEE3ERP) to decide who should process an abend, XDC/SE or L.E. XDC/SE either proceeds or returns (to RT/M) according to CEE3ERP's report.

Help Whatsnew S20 SCrolling

Displays of storage that have been mapped with source level maps can be significantly wider than most displays produced by previous versions of XDC and XDC/SE. Accordingly, it has finally become necessary for XDC/SE to support LEFT and RIGHT scrolling commands. So now XDC/SE does support LEFT and RIGHT commands, and the factory default PF-key assignments have been changed as follows:

KEY-SET	KEY#	WAS/NOW
SET-A	9th	was: SWAP now: T
SET-A	10th	was: T now: T BY
SET-A	11th	was: T BY now: T NXSEQ() NXSEQ(2) ...;GO
SET-B	9th	was: SWAP now: SWAP
SET-B	10th	was: T NXSEQ() NXSEQ(2) ...;GO now: LEFT -
SET-B	11th	was: T NXSEQ() NXSEQ(2) ...;GO now: RIGHT -

In other words, the tracing commands have been moved to PF-keys 9, 10, and 11 of the primary key set, and the "standard" SWAP, LEFT, and RIGHT commands have been assigned to the secondary key set. This may take some getting used to, but it seems likely that the tracing keys will be used much more frequently than the scrolling keys; therefore, it seemed reasonable to assign the tracing keys to the primary key set. Of course, you can always change your personal definitions to whatever you like. For more information, see HELP FULLSCREEN PFKEYS DFLTKEYS.

The SET WINDOW command has had to be changed to support setting default values for the LEFT/RIGHT scrolling commands. For more information, read about the SET WINDOW command in HELP WHATSNEW S20 COMMANDS.

XDC/SE™ S2.0 RELEASE GUIDE

Help Whatsnew S20 SOurcelevel

XDC/SE now offers source level debugging! This means that source program images can now be mapped to actively running program code and control blocks located in storage. So now when you display your code, you will now see:

- Your machine instructions exactly as they were written in the original source program, complete with commentary!
- Your commentary blocks.
- Your macro call statements.
- All relocatable symbols INCLUDING those defined by EQU statements.
- All statement labels INCLUDING those longer than 8 characters.

XDC/SE reads source level information from ADATA records produced by IBM's High Level Assembler. The ADATA records can be read for both csects and dsects. They can be read for csects located both within program objects and within load modules.

ADATA records can be read from any location that they can be found. Specifically, they can be read:

- From within program objects located within PDSE libraries.
- From within xobject files located in either sequential datasets or PDS/PDSE libraries.
- From SYSADATA files located in either sequential datasets or PDS/PDSE libraries.

When ADATA is read from XOBJECT or SYSADATA libraries, the member names must match the names of the csects whose ADATA is being read.

An XOBJECT file is an extended format object code file. The HL-Assembler produces an XOBJECT file when PARM='XOBJECT' is specified. The HL-Assembler places ADATA records into the XOBJECT file when PARM='ADATA,XOBJECT(ADATA)' is specified.

When building a program object, if IBM's Binder finds ADATA records in the XOBJECT files, then it will include those records into the program object. This is convenient for small programs, but for large programs the presence of ADATA substantially increases both the size needed for the PDSE into which the program object is placed and the amount of time needed by the Binder to create the program object. In fact both of these increases can be HUGE!

Generally, it is better to keep the ADATA records in a separate SYSADATA file. The High Level Assembler produces a SYSADATA file when PARM='ADATA' is specified and a //SYSADATA DD card is present.

ADATA records are read by the MAP and DMAP commands. There are no new operands for these commands; their usage has not changed. If ADATA records are available to these commands, then they will build their maps from the ADATA information; otherwise, (as before) they will look for mapping information in SYM records.

When the module being read by the MAP or DMAP command is a program object, if that program object contains ADATA records, then the command will automatically find those records and read them in preference to SYM records. On the other hand, if the ADATA to be read is located within an XOBJECT file or SYSADATA file, then the SET MAPLIBS command needs to be used ahead of time to make the file available to the MAP or DMAP command.

The MAPLIBS list is a list of one or more sequential and partitioned data sets that may contain mapping information for use by XDC/SE. When a MAPLIBS list is available, then the MAP and DMAP commands will search the listed datasets and libraries (in the order listed) ahead of the program object or load module when building CSECT and DSECT maps.

When a MAPLIBS listed dataset (or PDS/PDSE member) is read by the MAP or DMAP command, a filtered extraction of that file is automatically cached by XDC/SE in case that same file needs to be read again in the future. These cache files are located in 31-bit storage. They can be displayed by the LIST CACHE command and forced purged by the DELETE CACHE command. They also will be purged automatically when the associated MAPLIBS dataset is removed from the MAPLIBS list by the DELETE

XDC/SE™ S2.0 RELEASE GUIDE

(Help Whatsnew S20 SOurcelevel)

MAPLIBS command or the SET MAPLIBS RESET command. They also will be purged automatically (on an LRU basis) whenever XDC/SE detects a free storage shortage. See HELP COMMANDS LIST CACHE for more information.

The MAPLIBS list is created by the SET MAPLIBS command, can be displayed by the LIST MAPLIBS command, and is deleted by the DELETE MAPLIBS command. It also can be saved (by the SET MAPLIBS command) into the session profile. See HELP COMMANDS SET MAPLIBS for more information.

Displays of storage that have been mapped with source level maps can be significantly wider than most displays produced by previous versions of XDC and XDC/SE. Accordingly, it has finally become necessary for XDC/SE to support LEFT and RIGHT scrolling commands. So now XDC/SE does support LEFT and RIGHT commands, and the factory default PF-key assignments have been changed to accommodate them. For more information, see HELP WHATSNEW S20 SCROLLING.

Help Whatsnew S20 Miscellaneous

The following additional changes have been made to XDC/SE:

- The maximum symbol length accepted by XDC/SE has been increased from 57 characters to 63 characters. This affects:
 - Symbols read from ADATA records created by IBM's High Level Assembler.
 - Symbols read from SYM records created by Tachyon Software's Cross Assembler.
 - Symbols created by XDC/SE's EQUATE command.
 - The names of dsect maps either read from ADATA or created via the cloning form of the DMAP command.
- The minimum abbreviations of format control operands appearing on the DISPLAY, FORMAT, SHOW, and WHERE commands has been increased from 2 characters to 3 characters. Such operands include:

ADDRESSES	ASCII	SOURCE	DECIMAL
OFFSETS	EBCDIC	OBJECT	HEXADECIMAL
		BOTH	
- When storage is mapped by a Binder map that has been created by the DMAP command, then when storage that falls within that map is displayed, the offsets of the displayed storage will be shown relative to csect boundaries instead of relative to the start of the map. Note, using the DMAP command (instead of the MAP command) to build a Binder map is useful for mapping load modules whose locations are not described by normal system control blocks. See HELP COMMANDS DMAP LOAD CSECTS for more information.
- When XDC/SE receives an attention signal while the user program is running, comprehensive information is now automatically displayed that explains what the attention routine's current environment is, and makes suggestions about how to resume debugging in the user program's environment. See HELP ATTENTIONS INUSERPGM for more information.

XDC/SE™ S2.0 RELEASE GUIDE

INDEX

Please note that this index is sorted according to the ASCII collating sequence, not EBCDIC. In particular, this means that digits sort in front of (not behind) alphabetic, and that only some special characters sort in front of alphabetic. Others sort behind alphabetic.

The word processing program that is used here supports only two levels of index entries: main topics and sub-topics. When a sub-topic entry says "**see major topics**", this indicates that you should look for the same index entry among the main topics.

ADATA	10, 15	FASMREGS	13
cached MAPLIB files	10, 16	HOOK (deleted)	13
DELETE MAPLIBS command	16	HOOKPLI (deleted)	13
LIST MAPLIBS command	16	ISPF410 (deleted)	13
load modules	15	ISPF430 (deleted)	13
MAPLIBS list	15	ISPF4XX	13
program objects	15	QUICKREF	13
program objects, performance issues	15	REFRESH command	11
SET MAPLIBS command	16	REGPTRS	13
SYSADATA ddname	15	commands	
SYSADATA files	15	AT	8
XOBJECT files	15	ATX	8
ADDRESSES operand		COPY	8
DISPLAY command	16	DELETE CACHE	10
FORMAT command	16	DELETE MAPLIBS	11, 16
SHOW command	16	DELETE SLISTLIBS	12
WHERE command	16	deleted	12
ASCII operand		DISPLAY	8, 16
DISPLAY command	16	DMAP	10, 16
FORMAT command	16	DMAP command	15
SHOW command	16	END	9
WHERE command	16	EQUATE	16
Assembler, High Level	(See High Level Assembler)	FIND	9
associated commands		FORMAT	9, 16
WATCHs	7	HDEFERRED	10
AT command	8	LEFT	10, 14
store protected storage	8	line commands	(see major topics)
attentions		LIST AREGS	10
intercepting the user program	16	LIST ARn	10
ATX command	8	LIST CACHE	10
store protected storage	8	LIST EPSW	10
Binder maps		LIST MAPLIBS	11, 16
loading as a dsect	16	LIST PSW	10
BOTH operand	9	LIST REGS	10
DISPLAY command	16	LIST Rn	10
FORMAT command	9, 16	LIST SLISTLIBS	12
SET FORMAT command	9, 11	MAP	10
SHOW command	16	MAP command	15
WHERE command	16	REFRESH	11
branch mnemonics	9	RIGHT	10
breakpoints		RIGHT command	14
conditional expressions	5	SET BREAKPOINT	11
disabling	6	SET FORMAT	9, 11
cached MAPLIB files		SET MAPLIBS	11, 16
DELETE CACHE command	10, 16	SET SLISTLIBS	13
DELETE MAPLIBS command	10, 16	SET WINDOW	11
LIST CACHE command	10, 16	SHOW	11, 16
purging	16	SMAP	13
SET MAPLIBS RESET command	16	TRACE	12
CEE3ERP load module		TRACE S	13
Language Environment	14	TRAP	12
command files	13	WHERE	16
EREGPTRS	13	ZAP	12

XDC/SE™ S2.0 RELEASE GUIDE

(Index)

conditional expressions		
FALSE constant	5	
TRUE constant	6	
WATCHs	7	
control blocks		
SDWA	9	
COPY command	8	
ddnames		
SYSADATA	15	
DECIMAL operand		
DISPLAY command	16	
FORMAT command	16	
SHOW command	16	
WHERE command	16	
deferred HOOKs	10	
DELETE CACHE command	10	
cached MAPLIB files	10, 16	
DELETE MAPLIBS command	11	
ADATA records	16	
cached MAPLIB files	10, 16	
DELETE SLISTLIBS command	12	
deleted commands	12	
DISPLAY command	8, 16	
ADDRESSES operand	16	
ASCII operand	16	
BOTH operand	16	
DECIMAL operand	16	
EBCDIC operand	16	
fetch protection indicator	8	
HEXADECIMAL operand	16	
linecount operand	9	
OBJECT operand	16	
OFFSETS operand	16	
SOURCE operand	16	
store protection indicator	8	
DMAP command	10, 16	
ADATA records	10, 15	
maximum symbol name length	16	
source level maps	10	
DUMP operand		
END command	9	
e-mail		(see internet)
EBCDIC operand		
DISPLAY command	16	
FORMAT command	16	
SHOW command	16	
WHERE command	16	
END command	9	
DUMP operand	9	
EQUATE command	16	
maximum symbol name length	16	
EREGPTRS command files	13	
FALSE constant		
conditional expressions	5	
WATCHs	6, 7	
FASMREGS command file	13	
fetch protected storage		
indicator	8	
FIND command	9	
flags		
VRANODAE	9	
FORMAT command	9, 16	
ADDRESSES operand	16	
ASCII operand	16	
BOTH operand	9, 16	
context reactive branch mnemonics	9	
DECIMAL operand	16	
EBCDIC operand	16	
HEXADECIMAL operand	16	
linecount operand	9	
OBJECT operand	9, 16	
OFFSETS operand	16	
SOURCE operand	9, 16	
X-quote framing of hex displacements	9	
FORMAT operand		
LIST EPSW command	10	
LIST PSW command	10	
FTP address		(see internet)
HDEFERRED command	10	
HEXADECIMAL operand		
DISPLAY command	16	
FORMAT command	16	
SHOW command	16	
WHERE command	16	
High Level Assembler		
SYSADATA files	15	
XOBJECT files	15	
home page		(see internet)
HOOK command file (deleted)	13	
HOOKPLI command file (deleted)	13	
HOOKs		
deferred	10	
HORIZONTAL operand		
SET WINDOW command	11	
internet		
e-mail address	ii	
FTP address	ii	
home page	ii, iii	
web address	ii	
ISPF410 command file (deleted)	13	
ISPF430 command file (deleted)	13	
ISPF4XX command file	13	
K line command	12	
Language Environment		
CEE3ERP load module	14	
handshaking	14	
LEFT command	10, 14	
PF-keys	10	
legal statements		
trademark notice	iii	
usage warning	ii	
line commands		
K	12	
X	12	
linecount operand		
DISPLAY command	9	
FORMAT command	9	
LIST AREGS command	10	
LIST ARn command	10	
LIST CACHE command	10	
cached MAPLIB files	10, 16	
LIST EPSW command	10	
FORMAT operand	10	
LIST MAPLIBS command	11	
ADATA records	16	
LIST PSW command	10	
FORMAT operand	10	
LIST REGS command	10	
LIST Rn command	10	
LIST SLISTLIBS command	12	

XDC/SE™ S2.0 RELEASE GUIDE

(Index)

load modules		
CEE3ERP	14	
MAP command	10	
ADATA records	10, 15	
source level maps	10	
MAPLIB lists		
cached MAPLIB files	10, 16	
MAPLIBS list		
ADATA records	15	
DELETE MAPLIB command	16	
LIST MAPLIB command	16	
saving in the profile	16	
SET MAPLIB command	16	
maps		
MAPLIBS list	15	
source level	15	
OBJECT operand	9	
DISPLAY command	16	
FORMAT command	9, 16	
SET FORMAT command	9, 11	
SHOW command	16	
WHERE command	16	
OFFSETS operand		
DISPLAY command	16	
FORMAT command	16	
SHOW command	16	
WHERE command	16	
operands		
ADDRESSES	16	
ASCII	16	
BOTH	9, 11, 16	
DECIMAL	16	
DUMP	9	
EBCDIC	16	
FORMAT	10	
HEXADECIMAL	16	
HORIZONTAL	11	
OBJECT	9, 11, 16	
OFFSETS	16	
RESET	16	
SCROLL	11	
SOURCE	9, 11, 16	
VERTICAL	11	
PF-keys		
factory default: LEFT -	10, 14	
factory default: RIGHT -	10, 14	
factory default: SWAP	14	
factory default: T	14	
factory default: T BY	14	
factory default: T\NXSEQ() NXSEQ(2) ...;GO	14	
scrolling	10, 14	
tracing	10, 14	
point and shoot		
registers	10	
point-and-shoot commands		
QUICKREF command file	13	
profile		
MAPLIBS list	16	
program objects		
ADATA records	15	
performance issues	15	
QUICKREF command file	13	
point-and-shoot commands	13	
READ command		
REFRESH command	11	
REFRESH command	11	
command files	11	
READ command	11	
registers		
point and shoot	10	
zapping	10	
REGPTRS command files	13	
RESET operand		
SET MAPLIBS command	16	
RIGHT command	10, 14	
PF-keys	10	
SCROLL operand		
SET WINDOW command	11	
scrolling	14	
LEFT command	10	
PF-keys	14	
PF-keys command	10	
PF=keys	14	
RIGHT command	10	
SDWA control block		
Variable Recording Area	9	
SET BREAKPOINT command	11	
SET FORMAT command	9, 11	
BOTH operand	9, 11	
OBJECT operand	9, 11	
SOURCE operand	9, 11	
SET MAPLIBS command	11	
ADATA records	16	
RESET operand	16	
SET MAPLIBS RESET command		
cached MAPLIB files	16	
SET SLISTLIBS command	13	
SET WINDOW command	11	
HORIZONTAL operand	11	
SCROLL operand	11	
VERTICAL operand	11	
SHOW command	11, 16	
ADDRESSES operand	16	
ASCII operand	16	
BOTH operand	16	
DECIMAL operand	16	
EBCDIC operand	16	
HEXADECIMAL operand	16	
OBJECT operand	16	
OFFSETS operand	16	
SOURCE operand	16	
SMAP command	13	
source level maps	10, 15	
ADATA records	15	
source modules		
SRCONLHG	iii	
SOURCE operand	9	
DISPLAY command	16	
FORMAT command	9, 16	
SET FORMAT command	9, 11	
SHOW command	16	
WHERE command	16	
SRCONLHG source modules	iii	
storage		
fetch protection indicator	8	
store protected storage	12	
store protection indicator	8	
storage alteration trace	6	
disadvantages	6	
stop after	6	

XDC/SE™ S2.0 RELEASE GUIDE

(Index)

stop before	6
store protected storage	
AT command	8
ATX command	8
indicator	8
TRACE command	12
TRAP command	12
ZAP command	12
symbols	
maximum name length	16
SYSADATA ddname	15
SYSADATA files	
ADATA records	15
PDS/PDSE libraries	15
TRACE command	12
store protected storage	12
TRACE S command	13
tracing	
PF-keys	14
storage alteration trace	6
trademarks notice	iii
TRAP command	12
store protected storage	12
TRUE constant	
conditional expressions	6
usage warning	ii
User Communications Interface exit routine	
SRCONLHG source module	iii
XDCONLHG	iii
VERTICAL operand	
SET WINDOW command	11
VRANODAE flag	9
WATCH	7
associated commands	7
conditional expressions	7
FALSE constant	6, 7
multiple	7
web address	(see internet)
WHERE command	16
ADDRESSES operand	16
ASCII operand	16
BOTH operand	16
DECIMAL operand	16
EBCDIC operand	16
HEXADECIMAL operand	16
OBJECT operand	16
OFFSETS operand	16
SOURCE operand	16
X line command	12
XDCONLHG exit routine	iii
XOBJECT files	15
ADATA records	15
PDS/PDSE libraries	15
ZAP command	12
exclusive OR	12
store protected storage	12
zapping	
registers	10