



**z/XDC<sup>®</sup>**  
**RELEASE GUIDE**

**z/XDC<sup>®</sup> Release z1.8**  
**for z/OS**

David B. Cole

# **z/XDC® z1.8 RELEASE GUIDE**

## **PREFACE**

### **PROPRIETARY LEGEND**

z/XDC® 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 The program and documentation may not be published, reproduced, distributed, or made available to third parties for any purpose without the expressed written permission of Owner; however, a reasonable number of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 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!** z/XDC® 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®** family of 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 **928-771-2003**  
FAX:             **928-771-2005**  
E-Mail:          [sales@colesoft.com](mailto:sales@colesoft.com)  
Home Page:      <http://www.colesoft.com>

Our Technical Support contacts are:

Phone:           **540-456-8210**  
FAX:             **540-456-6658**  
E-Mail:          [techsupt@colesoft.com](mailto:techsupt@colesoft.com)  
Home Page:      <http://www.colesoft.com>  
FTP site:         <ftp.colesoft.com>

# **z/XDC® z1.8 RELEASE GUIDE**

(Preface)

Our Customer Services contacts are:

Phone:           **540-456-8210**  
FAX:             **540-456-6658**  
E-Mail:          [support@colesoft.com](mailto:support@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 z/XDC.
- 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 z/XDC.
- Links permitting existing customers to download a full set of z/XDC's documentation.
- An order form for obtaining an upgrade of XDC to its current version (z/XDC) and release (z1.8).

## **TRADEMARKS**

**TFS™**, **XDC-TFS™**, **CDF™**, **XDC-CDF™**, and **FASM™** are trademarks of Cole Software, LLC. **Extended Debugging Controller®**, **XDC®**, **XDC-SE®**, and **z/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.

## **ADDITIONAL MANUALS**

z/XDC customers may make as many copies of this manual as they feel is necessary for the legitimate use of z/XDC within their organization. Existing customers may download from our web site ([www.colesoft.com](http://www.colesoft.com)) printable copies of all of z/XDC's manuals. Each manual is available in PDF format.

In addition, all manuals (except the Installation Guide) can be printed directly from within z/XDC itself. To print your own set of manuals, start an z/XDC 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 Z18;SET PRINT CLOSE
```

Alternatively, you also can print these manuals by issuing z/XDC's **READ** command to run the **MANUALS** member of z/XDC's script library. Example: **READ DBCOLE<sup>1</sup>.XDCZ18.XDCCMDS(MANUALS)**.

---

<sup>1</sup>The library's high level qualifier may be different at your data center. Please ask your Systems Programmer.

# ***z/XDC<sup>®</sup> z1.8 RELEASE GUIDE***

(Preface)

You also may print a **Quick Reference** for z/XDC by issuing z/XDC's **READ** command to run the QUICKREF member of z/XDC's script library. Example: **READ DBCOLE.XDCZ18.XDCCMDS(QUICKREF)**.

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

# **z/XDC® z1.8 RELEASE GUIDE**

## **CONTENTS**

<b>PREFACE</b> .....	<a href="#"><u>ii</u></a>
<b>PROPRIETARY LEGEND</b> .....	<a href="#"><u>ii</u></a>
<b>CONTACTING COLE SOFTWARE</b> .....	<a href="#"><u>ii</u></a>
<b>TRADEMARKS</b> .....	<a href="#"><u>iii</u></a>
<b>ADDITIONAL MANUALS</b> .....	<a href="#"><u>iii</u></a>
<b>CONTENTS</b> .....	<a href="#"><u>v</u></a>
<b>INTRODUCTION</b> .....	<a href="#"><u>1</u></a>
A Roadmap .....	<a href="#"><u>1</u></a>
<b>Online Help Panels</b> .....	<a href="#"><u>3</u></a>
Help Whatsnew .....	<a href="#"><u>3</u></a>
Help Whatsnew Z18 .....	<a href="#"><u>3</u></a>
Help Whatsnew Z18 AData .....	<a href="#"><u>5</u></a>
Help Whatsnew Z18 ARTifacts .....	<a href="#"><u>5</u></a>
Help Whatsnew Z18 Breakpoints .....	<a href="#"><u>6</u></a>
Help Whatsnew Z18 CDf .....	<a href="#"><u>6</u></a>
Help Whatsnew Z18 COMmands .....	<a href="#"><u>6</u></a>
Help Whatsnew Z18 CONsoles .....	<a href="#"><u>11</u></a>
Help Whatsnew Z18 DDnames .....	<a href="#"><u>12</u></a>
Help Whatsnew Z18 DUmps .....	<a href="#"><u>13</u></a>
Help Whatsnew Z18 Equates .....	<a href="#"><u>13</u></a>
Help Whatsnew Z18 FOrmatting .....	<a href="#"><u>13</u></a>
Help Whatsnew Z18 FRr .....	<a href="#"><u>15</u></a>
Help Whatsnew Z18 MACros .....	<a href="#"><u>15</u></a>
Help Whatsnew Z18 MAPlibs .....	<a href="#"><u>16</u></a>
Help Whatsnew Z18 Onlinehelp .....	<a href="#"><u>16</u></a>
Help Whatsnew Z18 POintandshoot .....	<a href="#"><u>17</u></a>
Help Whatsnew Z18 PRofiles .....	<a href="#"><u>18</u></a>
Help Whatsnew Z18 Quickstart .....	<a href="#"><u>20</u></a>
Help Whatsnew Z18 Rexx .....	<a href="#"><u>20</u></a>
Help Whatsnew Z18 SEcurity .....	<a href="#"><u>20</u></a>
Help Whatsnew Z18 SRbmode .....	<a href="#"><u>23</u></a>
Help Whatsnew Z18 SToragescrolling .....	<a href="#"><u>24</u></a>
Help Whatsnew Z18 Xdccall .....	<a href="#"><u>24</u></a>
Help Whatsnew Z18 Z9 .....	<a href="#"><u>24</u></a>
Help Whatsnew Z18 MIscellaneous .....	<a href="#"><u>25</u></a>
Help Whatsnew Z18 Incompatibilities .....	<a href="#"><u>26</u></a>
Help Whatsnew Z18 Incompatibilities Danelen .....	<a href="#"><u>26</u></a>
Help Whatsnew Z18 Incompatibilities Exits .....	<a href="#"><u>27</u></a>
Help Whatsnew Z18 Incompatibilities Find .....	<a href="#"><u>27</u></a>
Help Whatsnew Z18 Incompatibilities PFkeys .....	<a href="#"><u>27</u></a>
Help Whatsnew Z18 Incompatibilities POintandshoot .....	<a href="#"><u>28</u></a>
Help Whatsnew Z18 Incompatibilities Retrieve .....	<a href="#"><u>28</u></a>
Help Whatsnew Z18 Incompatibilities SEcurity .....	<a href="#"><u>28</u></a>
Help Whatsnew Z18 Incompatibilities SETbang .....	<a href="#"><u>28</u></a>
Help Whatsnew Z18 Incompatibilities SHow .....	<a href="#"><u>29</u></a>
Help Whatsnew Z18 Incompatibilities XDCAcif .....	<a href="#"><u>29</u></a>
Help Whatsnew Z18 Incompatibilities XDCCPanel .....	<a href="#"><u>29</u></a>
<b>INDEX</b> .....	<a href="#"><u>31</u></a>

# ***z/XDC<sup>®</sup> z1.8 RELEASE GUIDE***

(Contents)

# **z/XDC® z1.8 RELEASE GUIDE**

## **INTRODUCTION**

Cole Software has pursued the goal of making z/XDC'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:

- **z/XDC® User Guide**  
Contains comprehensive tutorials about the many features and capabilities of z/XDC.
- **z/XDC® Commands**  
Contains the detailed syntax, usage descriptions, and examples of all of z/XDC's commands.
- **z/XDC® Messages**  
Contains descriptions of all of the messages that can be issued by z/XDC and all of its various components.
- **z/XDC® z1.8 Release Guide**  
Contains a history of all changes and upgrades made in the current release of z/XDC.
- **z/XDC® Quick Reference**  
Contains brief lists of z/XDC 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.
- The PDF copies of the printed manuals can be searched using typical PC-style searching commands.
- "Release Guides" for older versions and releases of z/XDC 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-sequential. 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 z/XDC user, then start with the **z/XDC® z1.8 Release Guide**. This will tell you what's new in this release of z/XDC. 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.

For new users, turn to the **z/XDC® 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 sequentially. 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

# **z/XDC<sup>®</sup> z1.8 RELEASE GUIDE**

## **(Introduction)**

information about whether and to what extent you may have to modify your program in order to use z/XDC.

- 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 z/XDC that permits user terminals to connect to debugging sessions for background jobs.

For z/XDC command information, turn to the **z/XDC<sup>®</sup> 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 z/XDC'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 800-XDC-5150 (USA: 928-771-2003). 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.

# **z/XDC® z1.8 RELEASE GUIDE**

## **Online Help Panels**

### **Help Whatsnew**

XDC'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 users who want a concise summary of what has changed and a road map of where to look for more specific information.

```
z/XDC    z1.8 - (09/06) Major changes:
          - Support for debugging SRB mode programs
          - Support for execution z/XDC as an FRR
          - Security rules support changes
          - Storage scrolling
          - Macro expansion display suppression
          - Quick Start in TSO or under XDCCALL
          - REXX commands support
          - New Z9 machine instructions support

z/XDC    z1.7 - (02/06) Beta release for z1.8
z/XDC    z1.6 - (11/04) Major changes:
          - Support for HL-ASM R1.5's ADATA
          - Support for z/OS R1.6's ALRF
          - Protected-storage protection support

z/XDC    z1.3 - (05/04) Autocloning, complete program object support, etc.
z/XDC    z1.2 - (10/03) Z/Architecture support (64-bit addressing, etc.)
XDC/SE   S2.0 - (12/00) Incremental changes implemented via maintenance.
XDC/SE   S2.0 - (08/00) New release: Source Level Debugging Support!
XDC/SE   S1.0 - (11/98) New version! PDS/E support! XMS Support! Etc.
XDC      X3.3 - (10/97) Incremental fixes and additions
XDC      X3.2 - (12/96) Incremental fixes and additions
XDC      X3.1 - (04/95) Beta-test release of X3.2
XDC      X3.0 - (06/94) MVS/ESA support
```

### **Help Whatsnew Z18**

**Support fo ESA/390 architecture has been dropped!** Starting with this release, z/Architecture is required.

z/XDC z1.8 includes all maintenance fixes to z1.6 and the following additional changes. For detailed information, you can select the following topics directly, or you can use HELP \*NEXT to proceed sequentially. Use HELP \*FORWARD to skip.

- ADATA** - Some improvements have been made to the display of storage formatted via ADATA maps.
- ARTIFACTS** - Transient structures that exist only during z/XDC's processing are now flagged as ARTIFACTs when they show up within various z/XDC displays.
- BREAKPOINTS** - Conditional expressions can now be nested and compounded.
- CDF** - Mixed case password support has been added.

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18)**

- COMMANDS** - Several z/XDC commands have been either added or changed.
- CONSOLES** - Changes have been made to z/XDC's support of System Operator Consoles, both in the way in which messages are routed to consoles, and in the management of consoles as a user interface for debugging.
- DDNAMES** - Support for several new ddnames has been added.
- DUMPS** - When an unexpected abend occurs, z/XDC can now be directed to get a dump itself (instead of relying upon RTM to do it).
- EQUATES** - Several new built-in equates have been defined.
- FORMATTING** - Several improvements have been made in the way in which z/XDC formats and displays storage.
- FRR** - z/XDC can now be used as an FRR in both task mode and SRB mode programming.
- MACROS** - The display of macro expansions can now be suppressed.
- MAPLIBS** - Support has been added for providing ADATA libraries via an xxxMAPLB ddname.
- ONLINEHELP** - New topics have been added to Online Help.
- POINTANDSHOOT** - The way in which Point-and-Shoot commands work has been changed.
- PROFILES** - Several changes have been made both to the factory default profile and to profiles in general.
- QUICKSTART** - Support has been added to z/XDC's Startup Panel and to the XDCCALL program to allow a debugging session to bypass the initial trap to z/XDC and to start the user's program running immediately.
- REXX** - Support has been added to z/XDC's User Commands Exit Interface permitting user commands to be written in REXX.
- SECURITY** - Some changes have been made to z/XDC's support for System Security rules. **IMPORTANT!** Some of these changes create incompatibilities with prior versions and releases of XDC.
- SRBMODE** - z/XDC now can be used to debug code that is running in SRB mode.
- STORAGESCROLLING** - z/XDC now can "scroll" downwards through storage (sort of).
- XDCCALL** - Several changes have been made to the XDCCALL utility.
- Z9** - Recognition has been added for the machine instructions that are new with the Z9 processor.
- MISCELLANEOUS** - Several additional minor changes have been made.
- INCOMPATIBILITIES** - Because of various changes in this release, several special recommendations need to be made and a few incompatibilities have arisen.

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18)**

### **Help Whatsnew Z18 AData**

The following improvements have been made relating to the display of storage that is formatted by ADATA maps:

- New format control operands have been implemented on the **FORMAT**, **WHERE**, **FIND** and **SET FORMAT** commands that affect whether or not macro expansions are displayed. The operands are:
  - **HIDEMCODE**: This causes all macro expansions to be suppressed in formatted displays.
  - **SHOWMCODE**: This causes all macro expansions to be included in formatted displays.
  - **CURRENTMCODE**: This causes all macro expansions to be suppressed **except** if the macro expansion currently being displayed includes the error level or retry level PSW address. In this case, the expansion is shown.

For more information, see **HELP MAPS ADATA MACROS**.

- When generating ADATA-formatted displays, z/XDC will now honor the Assembler's current **MCALL/NOMCALL** setting. For macro expansions in which **NOMCALL** is in effect, z/XDC will suppress displays of macro call statements issued from within the macro expansion. Such statements will be displayed only from macro expansions in which **MCALL** is in effect.

### **Help Whatsnew Z18 ARtifacts**

During its normal processing, it is inevitable that z/XDC will create system control blocks and structures that will show up in various displays, but that do not exist whenever the user's program is in control. In other words, these control blocks and structures are created whenever z/XDC receives control and are purged whenever z/XDC returns control to the user program. These transient structures are **artifacts** of the debugging session.

To help bring even more clarity to a murky world, whenever an artifact shows up within a display produced by z/XDC, the label **(ARTIFACT)** will now be appended to the display to show clearly that what is being shown will not exist whenever the user program is in control.

The following displays are affected:

#### **LIST RBS**

Usually, when the current task's request blocks queue is displayed, the two newest displayed RBs are artifacts of the debugging session. They are created by the System in order to pass control to z/XDC. So whenever z/XDC returns control to the user program (**TRACE** and **GO** commands), they will be purged, so the user program can never see them. These two RBs will now be labeled as **ARTIFACTS**.

#### **LIST ESTAES**

#### **LIST SCBS**

Whenever, z/XDC receives control, it issues an **ESTAEX** macro to protect itself from both intentional and accidental abends that may occur. Consequently, whenever the **LIST ESTAES** command (aka **LIST SCBS** command) is used to display the current task's STAE Control Block queue, z/XDC's internal **ESTAEX** will show up in the display as the newest SCB queued. This SCB will now be labeled as an

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 ARtifacts)**

ARTIFACT.

### **LIST LSTACK**

Whenever, z/XDC receives control, it issues a BAKR to save its caller's execution time registers and other state data. Consequently, whenever the LIST LSTACK command is issued to display the current linkage stack, the newest state entry will be z/XDC's own. This LSE will now be labeled as an ARTIFACT.

## **Help Whatsnew Z18 Breakpoints**

The syntax of conditional expressions (for setting up conditional traps and traces) has been substantially improved. The expressions can now be compounded and nested. Multiple tests can now be performed within a single expression, and the results of those tests can now be AND/OR/XOR'd together for making a stop/continue decision. See HELP COMMANDS SYNTAX BREAKPOINTS MULTIPLE for more information.

## **Help Whatsnew Z18 Cdf**

Support for mixed case passwords has been added to the CDF logon process. See HELP CDF LOGON for more information.

## **Help Whatsnew Z18 COMmands**

The following commands are either new to z/XDC z1.8, changed in z/XDC z1.8, newly documented in z/XDC z1.8 or deleted from z1.8.

**AT**  
**ATX**  
**ADEFERRED**  
**TRACE**  
**TRAP**  
**TDEFERRED**

The syntax of conditional expressions (for setting up conditional traps and traces) has been substantially improved. The expressions can now be compounded and nested. Multiple tests can now be performed within a single expression, and the results of those tests can now be AND/OR/XOR'd together for making a stop/continue decision. See HELP COMMANDS SYNTAX BREAKPOINTS MULTIPLE for more information.

### **COMMENTARY**

Several improvements have been made to the data entry window that is opened by the COMMENTARY command:

- The commentary data entry window is now line-numbered.
- UP and DOWN commands are now fully implemented within the window. (Previously, UP/DOWN [within this window] would not accept operands and always functioned as if the CURSOR operand had been given.)
- Pressing the ENTER key now causes the cursor to automatically move downwards one line in the window. When the cursor is located at the bottom of the window, a DOWN HALF command is automatically issued so as to open up more commentary data entry space.

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 COMmands)**

- When the commentary data entry window is ended and the commentary is displayed in the session log, that commentary is now framed by "start of commentary" and "end of commentary" messages.

For more information, see HELP COMMANDS COMMENTARY.

### **CONSOLE**

This command has been deleted in z/XDC z1.8. Its function has been replaced by a new command: **SET CONSOLE**. See HELP COMMANDS SET CONSOLE for more information.

### **DELETE MAPLIBS**

The DELETE MAPLIBS command now recognizes **QUIET** as an alias of the NOSHOW operand.

### **DOWN**

Whenever a DOWN command would cause the display window to move past the end of the session log, **and** if the last thing that had been displayed in the session log was a storage display, then z/XDC will now automatically issue a **DISPLAY** or **FORMAT** command (as appropriate) to append a display of the next chunk of storage to the bottom of the log, thereby extending the log and bringing that next chunk of storage into view. For more information, see HELP FULLSCREEN SCROLLING.

### **FIND**

Several changes have been made to the FIND command:

- The processing statistics displayed in response to the **STATISTICS** operand have been substantially improved.
- The **FORMAT** and **DISPLAY** operands now optionally accept suboperands that allow you to control the size of the display produced when the FIND command finds a match. Example: **FIND ... FORMAT=4**
- The FIND command no longer attempts to display storage when no match is found.
- If the FIND command should produce a formatted display of storage, then the **INSTRUCTION**, **DATA**, and **NOBIAS** operands now affect the display in the same way they would if used on a FORMAT command. (Previously, they only affected the attributes of equates produced via the EQUATENAMES= operand.)
- The FIND command now accepts three new operands that control whether formatted displays produced by the FIND command will include or suppress the display of code produced by macro expansions. The operands are **SHOWMCODE**, **HIDEMCODE** and **CURRENTMCODE**. For more information, see HELP MAPS ADATA MACROS.

For more information, see HELP COMMANDS FIND.

### **FORMAT**

The FORMAT command now accepts three new operands that control whether its displays will include or suppress the display of code produced by macro expansions. The operands are **SHOWMCODE**, **HIDEMCODE** and **CURRENTMCODE**. For more information, see HELP MAPS ADATA MACROS.

### **GO/GOT/GOX**

The GO, GOT and GOX commands have been recoded to accept multiple operands in any order. See HELP COMMANDS GO for more information. (Actually, this change was made

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 COMmands)**

several releases ago, it just hasn't been documented until now.)

### **LIST CONSOLES**

This command displays information about operator consoles that are defined to the Operating System. For more information, see HELP COMMANDS LIST CONSOLES.

### **LIST ESTAE**

This command has been renamed to **LIST ESTAES**.

### **LIST ESTAES**

#### **LIST SCBS**

Whenever the LIST ESTAES command is used to produce a display of the current task's STAE Control Block queue (SCBs), the newest displayed SCB will be labeled with **(ARTIFACT)** to show that it is an artifact of the debugging session. (See HELP WHATSNEW Z18 ARTIFACTS for more information.)

### **LIST FEATURES**

Support has been added to the LIST FEATURES command for displaying the presence or absence of the following hardware/software facilities:

- DAT Enhancement Facility 2 (i.e. the LPTEA instruction)
- Mixed Case Passwords Support
- Store Clock Fast Facility (i.e. the STCKF instruction)
- Store Facility List Extended Facility (i.e. the STFLE instruction)

### **LIST FORMAT**

The following changes have been made to the display produced by the LIST FORMAT command:

- The command now displays the CURRENT **SET LINES** value, the default display size for the FORMAT, DISPLAY, and WHERE commands used when z/XDC is running in line mode instead of fullscreen mode. See HELP COMMANDS SET LINES for more information.
- LIST FORMAT also displays the current **SHOWMCODE/HIDEMCODE/CURRENTMCODE** setting from the SET FORMAT command. (This controls whether or not formatted displays will show macro expansion code.)

### **LIST LOCKS**

This is a new command for displaying the status of locks held by the program being debugged. It shows what locks were held by the program and what locks will be reacquired when the program is resumed.

This command also shows if user program resumption will be prohibited by System Security due to locks having been lost.

Unfortunately, due to information limitations provided by RTM, this command's displays are useful only when z/XDC is running as an FRR. For all other types of recovery routines, locks held information is lost before z/XDC ever receives control. For more information, see HELP COMMANDS LIST LOCKS.

### **LIST LSTACK**

#### **LIST LSES**

A change has been made to the way in which this command is used. Now, when requesting a detailed display of a linkage stack entry, if one of the operands is

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 COMmands)**

the address of an existing LSE, RB or SCB, then an address of a TCB no longer has to be also given. Previously, omitting the TCB address was permitted only when the desired LSE resided on the current task's linkage stack. Now, the TCB address can be omitted no matter what task (and no matter what address space) the desired linkage stack entry resides in.

Another change: When the LIST LSTACK command is used to produce a summary display, an **LSE#0** equate is now generated for labeling a linkage stack's first header entry. Thus, even the display of empty linkage stacks will still produce at least one equate labeling the start of the stack. (2nd and subsequent header entries continue not to be labeled.)

Finally, whenever the LIST LSTACK command is used to produce a summary display of the current task's linkage stack, the newest displayed state entry will be labeled with **(ARTIFACT)** to show that it is an artifact of the debugging session. (See HELP WHATSNEW Z18 ARTIFACTS for more information.)

### **LIST RBS**

Whenever a LIST RBS command is used to produce a display of the current task's request block queue, the one or two newest displayed request blocks will be labeled with **(ARTIFACT)** to show that they are artifacts of the debugging session. (See HELP WHATSNEW Z18 ARTIFACTS for more information.)

### **LIST SECURITY**

This is a new command that displays certain aspects of z/XDC's security processing. For more information, see HELP COMMANDS LIST SECURITY.

### **LIST STATISTICS**

This command displays information about the GETMAINS and FREEMAINS that z/XDC issues internally during its processing. The report produced by this command has been enhanced to include information about 31-bit GETMAINS for which the System has returned 24-bit storage. For more information, see HELP COMMANDS LIST STATISTICS.

### **R**

The minimum abbreviation for the READ command remains **R** (regardless of other commands starting with R).

### **RETRIEVE cmd#**

When the RETRIEVE command (or its aliases, RETRY and RECALL) is given with an **unsigned** command number operand, that operand is now treated as a negative number. This means that the nth **prior** command is retrieved. (Previously, an unsigned operand meant the nth following command was retrieved.) For more information, see HELP COMMANDS RETRIEVE.

### **RETRIEVE LIST**

The **LIST** operand has finally been implemented for the RETRIEVE command. Using it causes a window to be displayed that shows all recently issued commands. You then can do any or all of several things.

- You can modify any of the retrievable commands.
- You can directly select a retrievable command for reexecution.
- You can purge retrievable commands from the list.

For more information, see HELP COMMANDS RETRIEVE.

### **REXX**

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 COMmands)**

This is an alias of the XRI command. See below for more information.

### **SET CONSOLE**

This is a new command that can be used to cause z/XDC to switch the debugging session's user interface back and forth between the user's terminal and an Operator's Console. See HELP COMMANDS SET CONSOLE for more information.

### **SET FORMAT**

The SET FORMAT command now accepts three new operands that control whether formatted displays produced by the FORMAT, WHERE, and FIND commands will include or suppress the display of code produced by macro expansions. The operands are **SHOWMCODE**, **HIDEMCODE** and **CURRENTMCODE**. For more information, see HELP MAPS ADATA MACROS.

### **SET BANG**

The factory default setting for this command has been changed from AMODE to **64BIT**. See HELP WHATSNEW Z18 INCOMPATIBILITIES SETBANG for more information.

### **SET LOCKS**

This is a new command that allows the user to change the set of locks that will be (re)acquired upon user program resumption. It is effective only when z/XDC is running as an FRR. For more information, see HELP COMMANDS SET LOCKS.

### **SET MAPLIBS**

The SET MAPLIBS command now recognizes **QUIET** as an alias of the NOSHOW operand.

### **SET SECURITY**

This is a new command for setting certain aspects of z/XDC's security processing. For more information, see HELP COMMANDS SET SECURITY.

### **SET WINDOW CREATE nn**

This command accepts a new operand (nn) by which you can explicitly specify the terminal display line at which a new watch window is to be created. (This makes it possible now to use the SET WINDOW CREATE command within a script.) For more information, see HELP COMMANDS SET WINDOW CREATEDELETE.

In conjunction with this, the factory default value for PF-key 13 has been changed to **SET WINDOW CREATE -** (i.e. a trailing hyphen has been added to the end of the definition). This makes it possible (but not required) to provide an operand from the command line prior to pressing the PF-key. For more information, see HELP FULLSCREEN PFKEYS DFLTKEYS SETBC.

### **SET WINDOW DELETE nn**

This command accepts a new operand (nn) by which you can explicitly specify the particular display window to be deleted from the display screen. For more information, see HELP COMMANDS SET WINDOW CREATEDELETE.

In conjunction with this, the factory default value for PF-key 1 has been changed to **SET WINDOW DELETE -** (i.e. a trailing hyphen has been added to the end of the definition). This makes it possible (but not required) to provide an operand from the command line prior to pressing the PF-key. For more information, see HELP FULLSCREEN PFKEYS DFLTKEYS SETA.

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 COMmands)**

### **SHOW**

The following changes have been made to the SHOW command.

- When SHOW is used to display z/XDC messages, multi-line messages are now displayed in their entirety. (Previously, only their first lines were displayed.) For more information, see HELP COMMANDS SHOW.
- SHOW no longer accepts the SOURCE, OBJECT, and BOTH operands. SHOW now unconditionally displays only formatted object code (SYM data), never source code (ADATA).

### **SYSINFO**

This is a new command script that produces displays regarding the release, maintenance level, features available and general information about z/XDC, the Operating System, and (sometimes) the user program. For more information, see HELP SCRIPTS SYSINFO.

### **TRACE**

### **TRAP**

### **TDEFERRED**

### **AT**

### **ATX**

### **ADEFERRED**

The syntax of conditional expressions (for setting up conditional traps and traces) has been substantially improved. The expressions can now be compounded and nested. Multiple tests can now be performed within a single expression, and the results of those tests can now be AND/OR/XOR'd together for making a stop/continue decision. See HELP COMMANDS SYNTAX BREAKPOINTS MULTIPLE for more information.

### **WHERE**

The WHERE command now accepts three new operands that control whether its displays will include or suppress the display of code produced by macro expansions. The operands are **SHOWMCODE**, **HIDEMCODE** and **CURRENTMCODE**. For more information, see HELP MAPS ADATA MACROS.

### **XRI**

### **REXX**

**XRI** stands for "XDC/REXX Interface". **XRI** is a new command that allows the user to run user written subcommands written in REXX. See HELP COMMANDS XRI for details.

## **Help Whatsnew Z18 CONsoles**

z/XDC's support for both sending messages to and receiving commands from System Operator Consoles has been completely rewritten. Now, whenever z/XDC issues WTOs, if the message being sent is a result of a command from a console, then that WTO will be routed back to the same console from which the command was issued. More specifically:

- **WTOs:** Whenever z/XDC receives a reply to any WTOR, it will automatically recognize the console from which that reply came, and it will route all response messages generated by that reply back to the issuing console.
- **MODIFY xxxCDF Commands:** z/XDC's Cross Domain Facility (CDF) will now recognize the console from which a START (S), STOP (P), or MODIFY (F) command came, and it

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 CONsoles)**

will route all response messages back to that console.

- **Debugging Sessions:** When z/XDC has been directed to connect the debugging session's user interface to a System Operator Console, z/XDC will automatically recognize the console from which a session command was issued, and it will route all response messages generated by that command back to the issuing console.

Further, z/XDC's prompt (a WTOR) for the next command will be routed to the same console from which the prior prompt's response came. In other words, the debugging conversation will automatically follow the user around, should he wander from one console to another.

### **4-Byte Console IDs**

Internally, z/XDC now routes its WTOs either by console name or by 4-byte console id number (whichever is appropriate).

### **EMCS Console Support**

z/XDC console routing now, for the first time, explicitly supports EMCS consoles (pseudo consoles created by software such as SDSF, IOF, and other programs). Commands issued from EMCS consoles (like from any other consoles) will have their responses routed back to those consoles.

### **Console Related Commands**

- The **CONSOLE** command has been removed. It no longer exists.
- A **SET CONSOLE** command has been implemented by which the debugging conversation can be explicitly switched from the user's terminal, to the Operator Console(s), and back again. See **HELP COMMANDS SET CONSOLE** for more information.
- A **LIST CONSOLES** command has been added by which the user can see what Operator Consoles are available.

## **Help Whatsnew Z18 DDnames**

Support for the following ddnames has been either added or changed. Select the hyperlinks for more detailed information. For complete information about all ddnames related to z/XDC, see **HELP DDNAMES**.

**XDCISxxx:** Notifies z/XDC, running in JES2 or CICS, what its clone name is.

**xxxMAPLB:** Allows a MAPLIBS list to be preallocated and provided by ddname.

**xxxQUICK:** Causes a debugging session to begin running the user's program immediately, without first trapping to a z/XDC display.

**xxxRENT8:** When debugging a nonauthorized reentrant program, this can be used to cause that program to be loaded into key 8 storage (instead of key 0 storage).

**xxxSDUMP:** When z/XDC fails due to an unexpected abend, if z/XDC had been running authorized, then this ddname causes z/XDC to attempt to take a dump by issuing the SDUMPX macro (instead of relying upon RTM to do it).

# z/XDC® z1.8 RELEASE GUIDE

## (Help Whatsnew Z18 DDnames)

**xxxTRACE:** Online Help has been written to document this ddname. xxxTRACE can be used to help diagnose problems that might arise in z/XDC's communications with a user's terminal.

**xxxTRSAF:** This can be used to activate a z/XDC related security calls trace during the startup of a debugging session (prior to it being possible to issue a SET SECURITY TRACE command).

**xxxXRI:** This points to a library of REXX programs to be used as user written z/XDC commands.

## Help Whatsnew Z18 DUMps

Normally, when z/XDC fails due to an unexpected abend, it produces a summary dump and then percolates the abend back to RTM with SDWAREC=1 (requesting that RTM produce a dump). However, if z/XDC's internal recovery routine detects the presence of a //xxxSDUMP allocation, and if z/XDC is running authorized, then it will issue an SDUMPX macro before percolating. For more information, see HELP SUPPORT DUMPS.

## Help Whatsnew Z18 Equates

Two new sets of built-in equates have been created. They are @RWn equates and @ERWn equates. The @RWn equates label the locations pointed to by the 64-bit retry level general registers, but **only** when they point to locations above the bar.

Similarly, the @ERWn equates label the locations pointed to by the 64-bit **error** level general registers.

For more information, see HELP EQUATES BUILTIN REGISTERS.

## Help Whatsnew Z18 FOrmatting

Several improvements have been made to the way in which z/XDC formats and displays storage:

### Formatting Consistency

There are a multitude of options that affect the formatting of storage. Their default values can be set via the SET FORMAT command, and they can be individually overridden by operands on the DISPLAY, FIND, FORMAT, WHERE and various LIST commands. (See HELP COMMANDS SET FORMAT for details.)

The above is old news. The **new news** is, when one display of storage is generated via some action on or related to a prior display of storage, then the formatting options used with that prior display are reused to control the formatting of the new display.

Here are examples of such derivative displays:

#### - Line Commands

When a D, F, or W line command is issued from a line that is displaying

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 FOrmatting)**

storage, the formatting options associated with that line are used for generating the new display.

### **- Point and Shoot Commands**

When a point-and-shoot command is issued from within a line that is displaying storage, the formatting options associated with that line are used for generating the new display.

### **- Downward Scrolling Through Storage**

When a DOWN command is used to scroll through storage (see HELP WHATSNEW Z18 STORAGE\_SCROLLING), the newly generated storage displays will have the same formatting controls as the storage display from which the DOWN command was issued.

### **- FORMAT and DISPLAY Commands Without Operands**

When a FORMAT or DISPLAY command is given without operands, then the formatting controls used will be the same formatting controls as had been used by the most recently issued FORMAT or DISPLAY command (respectively).

## **Macro Expansion Displays**

New operands for the FORMAT, FIND, SET FORMAT and WHERE commands have been implemented that control whether or not macro expansions are displayed or suppressed (in storage formatted by an ADATA map). These operands are:

- **HIDEMCODE** to suppress macro expansion displays.
- **SHOWMCODE** to show those displays.
- **CURRENTMCODE** to show only those macro expansions into which the retry level or error level PSW falls.

For more information, see HELP MAPS ADATA MACROS.

## **MCALL Support**

When formatting storage under the control of an ADATA map, and the code being displayed was generated by a macro, z/XDC now checks for and honors the Assembler's MCALL setting at the current display point. When MCALL(NO) is in effect, images of inner macro call statements will not be produced.

## **Storage Offset Displays**

The location offsets, displayed down the lefthand side of storage displays, are no longer affected by the locations of z/XDC's built-in register equates (@Rn, @ERn, @RWn and @ERWn). The removal of this clutter hopefully will make the offset information more useful.

## **Storage Scrolling**

Whenever a DOWN command would cause the display window to move past the end of the session log, and if the last thing that had been displayed in the session log was a storage display, then z/XDC will now automatically issue a **DISPLAY** or **FORMAT** command (as appropriate) to append a display of the next chunk of storage to the bottom of the log, thereby extending the log and bringing that next chunk of storage into view. For more information, see HELP COMMANDS DOWN.

# **z/XDC<sup>®</sup> z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 FOrmatting)**

### **Help Whatsnew Z18 FRr**

A major enhancement for this release is the addition of support for allowing z/XDC to be setup to run as an FRR type recovery routine (Functional Recovery Routine). The FRR routine can be protecting a task mode program or an SRB routine. Either way, z/XDC will work. For detailed information, see HELP DEBUGGING FRR.

Every effort has been taken to make the use of z/XDC-as-an-FRR as noninvasive to the target program as possible. However, there is one major problem: z/XDC must release all locks held by the target program whenever z/XDC receives control! (This is unavoidable.)

Whenever, z/XDC returns control to the target program, it reacquires all prior locks before doing so; however, the consequences of this compromise are profound! For complete details, see HELP DEBUGGING FRR.

Two new commands and a new security rule have been implemented for this FRR support:

- **LIST LOCKS** displays the locks that were held by the routine being debugged and that will be reacquired when that routine is resumed. This command also shows if user program resumption will be prohibited by System Security due to locks having been lost.
- **SET LOCKS** allows you to alter the set of locks that will be reacquired upon resumption.
- A new security rule (XDC.LOSTLOCKS) has been implemented that allows the control of who will and will not be permitted to resume the execution of a program after locks have been lost (as a result of passing control to z/XDC in the first place).

**WARNING!** Unlike all other XDC related security rules, XDC will interpret NOT PROTECTED for this rule as being **ACCESS DENIED**. Therefore, if you wish to use z/XDC to debug programs that hold locks, this XDC.LOSTLOCKS rule will have to be created, and you will need to be given READ access to it. For more information, see HELP SECURITY LOSTLOCKS.

Note, Whenever z/XDC receives control and is able to detect that locks have been lost, it issues message DBC933W to alert you to this circumstance.

### **Help Whatsnew Z18 MACros**

Support has been added to the storage display formatting commands that allows you to suppress, if you wish, the display of code that was generated by macro expansions. Several commands have been changed to accept new operands:

- **HIDEMCODE** to suppress macro expansion displays.
- **SHOWMCODE** to show those displays.
- **CURRENTMCODE** to show only those macro expansions into which the retry level or error level PSW falls.

Those commands are:

```
FIND
FORMAT
SET FORMAT
```

# **z/XDC<sup>®</sup> z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 MACros)**

WHERE

For more information, see HELP MAPS ADATA MACROS.

## **Help Whatsnew Z18 MAPlibs**

Support has been added for providing ADATA libraries to z/XDC via the ddname xxxMAPLB. The ddname can be a concatenation of partitioned datasets or sequential datasets (but not both). For more information, see HELP MAPS ADATA MAPLIBS.

## **Help Whatsnew Z18 Onlinehelp**

In addition to the normal documentation changes required for this release, the following entirely new topics have been added to the Online Help.

### **HELP CDF RECOVERY**

Sometimes, when debugging your program via z/XDC's Cross Domain Facility (CDF), it can happen that your terminal will lose its connection to the debugging session without either VTAM or CDF being aware of it. When this occurs, reconnecting can be difficult. HELP CDF RECOVERY is a new topic that describes in detail how to recover from this problem.

### **HELP DDNAMES**

This topic has been added to document, in one place, all of the ddnames recognized by z/XDC.

### **HELP DEBUGGING CICS**

The Online Help discussing the use of z/XDC within CICS has been improved.

### **HELP DEBUGGING OTHERESTAES**

This topic has been renamed to HELP DEBUGGING **ESTAES**. (This, of course, affects the names of all subtopics of this topic as well.)

### **HELP FULLSCREEN TERMINALS PROBLEMS**

This topic has been added to explicitly document use of the //xxxTRACE DDname to obtain a terminal communications trace (for troubleshooting problems z/XDC might be having with communication to user terminals).

### **HELP MAPS MODULEMAPDSECTS**

The Online Help discussing the management of Binder maps and csect maps loaded **as dsects** has been significantly improved.

### **HELP PROFILES**

The Online Help discussing the creation and management of session profiles has been moved out from being a subtopic of the FULLSCREEN topic. It is now a main topic named HELP PROFILES. (It was HELP FULLSCREEN PROFILE.) This change affects, of course, all subtopics of the PROFILES topic.

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 Onlinehelp)**

### **HELP SCRIPTS ISPF5XX**

This topic, discussing the ISPF5XX commands script, has been entirely rewritten.

### **HELP SUPPORT CONTACTUS**

The HELP SUPPORT topic has been redesigned. So the contact information, that had been in HELP SUPPORT, can now be found in HELP SUPPORT CONTACTUS.

### **HELP SUPPORT CREDITS**

This is a new topic that gives a brief outline of the history of z/XDC and gives credit to those people who have been and who continue to be important to the success of z/XDC and of Cole Software in general.

### **HELP SUPPORT DUMPS**

When you have a dump of a z/XDC problem, we may (or may not) ask you to send that dump to us. If we do want the dump, then you will have to compress it, convert it to a transmittable form and send it to our FTP site. HELP SUPPORT DUMPS is a new topic that explains in detail how to do that.

### **HELP XDCCLONES**

z/XDC has long had comprehensive support for having its name changed from XDC to pretty much anything else. The primary purpose for doing this is to allow the installation and simultaneous usage of multiple z/XDCs and have them operate with complete independence from each other.

Until now, the Online Help's documentation of this topic has been (to be polite) rather sketchy. So to address this deficiency, I have added a new topic named XDCCLONES, and I have made extensive changes to affected topics throughout the Online Help.

There are many additional new topics and changed that relate specifically to the other changes documented under HELP WHATSNEW Z18. References to those topics are made where appropriate.

## **Help Whatsnew Z18 POintandshoot**

Point-and-Shoot commands are 1-character and 2-character commands that are placed directly onto hex data within z/XDC's displays. They cause the data over which they are entered to be used as a pointer, and they result in the display of the storage being pointed to.

Previously, the width of the pointer data was limited by the width of the hex data field into which the pointer command was entered. That restriction has now been removed.

For example, if the following storage is displayed:

```
00A0CD80 8f 12345678 9ABCDEF0          *.....0*
```

Then previously, a "?" (31-bit indirect) placed on the "5" would result in the display of storage located at 00005678. Now, however, it will result in the display of location 56789ABC.

The relaxation of this limitation is useful when a formatted display of storage is

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 POintandshoot)**

fragmented in such a way that addresses within the displayed data are split onto two or more lines of the display. (This fragmentation can happen for many reasons including the placement of equates, the locations pointed to by registers, the locations of the ends of data areas, incorrectly defined or placed maps, use of the assembler ORG statement, etc. etc.)

For more information about Point-and-Shoot commands, see HELP FULLSCREEN POINTANDSHOOT.

## **Help Whatsnew Z18 PRofiles**

Several changes have been made regarding z/XDC's support of session profiles.

### **Online Help Changes**

The Online Help discussing the creation and management of session profiles has been moved out from being a subtopic of the FULLSCREEN topic. It is now a main topic named HELP PROFILES. (It had been HELP FULLSCREEN PROFILE.) This change affects, of course, all subtopics of the PROFILES topic.

### **Default Profile Changes**

The following changes have been made to z/XDC's factory default profile settings. They will become visible to you only if you issue:

- **PROFILE RESET:** To load z/XDC's hard-coded factory defaults.
- **PROFILE READ XDC TFS:** To load your Data Center's system-wide defaults. (maybe)

You may want to review these changes carefully and consider whether or not you want to change your own personal profiles to match some or all of these new settings.

#### **- PF-key 1:**

Was: SET WINDOW DELETE

Is: SET WINDOW DELETE -

This change causes command line content, if any, to be appended to the command (replacing the hyphen) before the command is issued. This allows a window to be deleted by window number instead of by cursor location.

#### **- PF-key 11:**

Was: T NXSEQ() NXSEQ(2) NXSEQ(3) NXSEQ(4) NXSEQ(5);GOT

Is: T NXSEQ() NXSEQ(2) NXSEQ(3) NXSEQ(4) NXSEQ(5) **NXSEQ(6);GOT**

This change supports a subroutine return into a vector that is up to **six** slots long (not just 5).

#### **- PF-keys 13 and 25:**

Was: SET WINDOW CREATE

Is: SET WINDOW CREATE -

This change causes command line content, if any, to be appended to the command (replacing the hyphen) before the command is issued. This allows a window to be created by row number instead of by cursor location.

#### **- PF-keys 21 and 33:**

Was: SWAP -

Is: SWAP **NEXT**

This change makes it easier to navigate amongst ISPF windows when more than two

# **z/XDC® z1.8 RELEASE GUIDE**

(Help Whatsnew Z18 PProfiles)

exist.

**- PF-keys 26:**

Was: SPLIT

Is: SPLIT **NEW**

This change makes it easy to create more than two ISPF windows. Also, it now matches PF-key 14's default setting (which has been "SPLIT NEW" since z1.6.)

**- Online Help PF-keys 9 and 21:**

Was: SWAP -

Is: SWAP **NEXT**

This change makes it easier to navigate amongst ISPF windows when more than two exist.

**- Default command line command retrieval list size:**

Was: 32 commands

Is: 256 commands

The new RETRIEVE LIST command makes it easier to navigate a large list of prior commands. So there was no reason to keep the retrieval list small.

**- The FIND command's initial storage display method:**

Was: Use a DISPLAY command

Is: Use a FORMAT command

A FORMAT-based display will more often be more useful than a DISPLAY-based display.

**- The default meaning of the exclamation point (!) indirect operator:**

Was: Sensitive to the current addressing mode

Is: Always loads a 64-bit wide address

Reasons:

- This change makes the meaning of "!" consistent with IBM's usage.
- More and more programmers are beginning to use the 64-bit addressing mode, so this meaning of "!" is becoming more and more useful to the programming community as a whole.
- An AMODE-sensitive indirect operation can still be achieved through use of the **~INDIRECT(AMODE)** built-in function. (Useful mainly when used in scripts.)

## **New Profiled Settings**

The SET FORMAT command has a new set of settings that relate to the display of macro expansions. They are SHOWMCODE, HIDEMCODE and CURRENTMCODE. (See HELP COMMANDS SET FORMAT for more information.) This setting is profiled, and the initial default is SHOWMCODE.

## **Changing Your Profile Safely**

One problem that sometimes occurs when people try to change their profile is that they wind up saving more changes than they expect. This occurs because there are **a lot** of profiled settings, and during a debugging session, settings may be changed without the user realizing it. Normally, this does not create problems because profiles are saved only explicitly, never automatically. Consequently, lurking profile changes do not get hardened unless and until a PROFILE SAVE command is issued. So when you intentionally save a profile, you may wind up saving more changes than you remember making.

It's easy to avoid this problem, though. All you have to do is issue commands in

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 PProfiles)**

the following sequence:

- First, issue a **PROFILE READ** command to insure that all lurking changes are cleared.
- Then issue the commands needed for making whatever profile changes you want to make.
- Finally, issue a **PROFILE SAVE** command to save your updated profile.

## **Help Whatsnew Z18 Quickstart**

The XDCCALL program now supports a "quick start" feature that allows the user to control whether or not z/XDC is given control before the user program starts up. This feature is controlled by the presence or absence of a dummy allocation named //xxxQUICK. For more information, see HELP XDCCALL DDNAMES.

z/XDC's Startup Panel (XDCCPANEL) in ISPF has been modified to take advantage of the quick start feature. The panel now has a **Quick Start?** field that allows the user to choose whether or not a quick start is to occur. For more information, see HELP DEBUGGING ISPF PANEL.

## **Help Whatsnew Z18 Rexx**

Support for user written REXX commands has been added to z/XDC's User Commands Exit Interface. This support permits users to write their own z/XDC commands using the REXX programming language.

To use this support, all you have to do is allocate your library of REXX execs to ddname //xxxXRI. Then run your exec by issuing the command: **XRI membername operands**, where:

- **XRI** is a new z/XDC command for running REXX subcommands.
- **membername** is the name of the library member that contains the REXX command that you want to run.
- and **operands** is whatever argument string (if any) that your REXX command may require.

For more information, see HELP REXX.

## **Help Whatsnew Z18 Security**

z/XDC has new security rules and new security related commands. Also, there are changes that have been made to z/XDC's support of existing System Security rules. In addition, it is now possible to display a trace of z/XDC's security calls, showing their resulting rulings.

**IMPORTANT!** The rule changes may create both current and future incompatibilities with existing Data Center security rule definitions! So please read carefully.

**The XDC.GZAP Rule**

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 SEcurity)**

**This rule is no longer supported!** All customers that still use this rule at their Data Centers **\*MUST\*** replace this rule with one or more **XDC.ZAP.COMMON.---** rules. (Note, an exact equivalent to the XDC.GZAP rule is **XDC.ZAP.COMMON.\***) For more information, see HELP SECURITY GZAP.

The **XDC.GZAP** rule is an ancient rule of limited value. In 1995, I implemented a more comprehensive set of rules (**XDC.ZAP.---**) for controlling zaps to storage, and I've been threatening to remove support for the **XDC.GZAP** rule ever since. Well, now's the time.

The **XDC.ZAP.---** rules allow you to more precisely target what you want to protect. They allow you to set up rules to protect specific storage subpools, specific load modules, or just general areas of storage.

But if you don't want to take advantage of the **XDC.ZAP.---** rule's greater power, and instead you just want to continue with the exact same functionality that the **XDC.GZAP** rule provided, then (as noted above) the rule named **XDC.ZAP.COMMON.\*** is exactly equivalent to the **XDC.GZAP** rule.

For more information about **XDC.ZAP.---** rules, see HELP SECURITY ZAP.

### **New Rules for controlling Zaps to Load Modules**

There are new formats for security rules that protect load modules and program objects. The keyword **.MODULE** has been added to the rule strings that z/XDC constructs for representing attempts to zap load modules. The old format for such rules was:

```
XDC.ZAP.COMMON.modulename.---  
XDC.ZAP.PRIVATE.modulename.---
```

**The new format is:**

```
XDC.ZAP.COMMON.MODULE.modulename.---  
XDC.ZAP.PRIVATE.MODULE.modulename.---
```

This rule change is being made to make it easier for Security Officers to create generic rules for protecting load modules categorically without also affecting protection for data areas.

For the time being, z/XDC will support both formats, but in a future release, support for the old format will be dropped. So we strongly recommend that your Data Center's Security Officer update your z/XDC security rules (if any) as soon as is reasonable.

For more information, see HELP SECURITY ZAP.

### **New Security Rules**

Two new security rules have been implemented. They are:

#### **XDC.REFEREE.ZAPMODULERULES**

This is a temporary rule used in permit/deny decisions for zapping load modules. During the conversion period, it is possible for new and old format security rules to give contradictory recommendations. So this "referee" rule is needed to determine, on a user by user basis, which recommendation to follow. See HELP SECURITY ZAP for more information.

#### **XDC.LOSTLOCKS**

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 Security)**

When z/XDC is being used to debug a routine that is holding locks, it is necessary that all such locks be released before z/XDC can suspend the routine and issue its displays. Then when the user issues a command to resume the routine's execution, all of the lost locks have to be reacquired. This is an **exceedingly dangerous** thing to do. Consequently, I felt that it was necessary to provide a security rule by which a Data Center can establish a policy by which it can control who may and may not be allowed to use z/XDC in this way. This XDC.LOSTLOCKS rule serves this purpose. When z/XDC receives control in a situation where it can determine whether or not locks have been lost, and if in fact locks have actually been lost, then if the user attempts to resume that program's execution, this rule will be checked to see whether or not that resumption will be permitted.

**WARNING!** Unlike all other XDC related security rules, XDC will interpret NOT PROTECTED for this rule as being **ACCESS DENIED**. Therefore, if you wish to use z/XDC to debug programs that hold locks, this XDC.LOSTLOCKS rule will have to be created, and you will need to be given READ access to it. For more information, see HELP HELP SECURITY LOSTLOCKS.

Note, whenever z/XDC receives control and is able to detect that locks have been lost, it issues message DBC933W to alert you to this circumstance.

### **New Security Related Commands**

Two new commands have been implemented that are related to security. They are:

- **SET SECURITY** for controlling certain security related behaviors of z/XDC.
- **LIST SECURITY** for displaying certain aspects of z/XDC's security processing.

For more information, see:  
HELP COMMANDS SET SECURITY  
HELP COMMANDS LIST SECURITY

### **Displaying a Security Trace**

Using either a keyword ddname (//xxxTRSAF) or the SET SECURITY command, it is now possible to cause z/XDC to display a report for each security call that it makes. The report shows (among other things):

- Why security is being called.
- The resource name that z/XDC wants the security system to rule upon.
- The specific rule (when available) that security looked up in order to make its ruling.
- The ruling that security has made.
- The final decision that z/XDC has made as a result of the security call.

For more information, see HELP SECURITY TRACE.

### **Online Help Changes and Improvements**

Several new topics have been added to the SECURITY section of the Online Help. They are:

#### **HELP SECURITY ZAP OLDRULES**

One of the major changes happening to z/XDC's security interface with this release is the changing of the format of the resource names that z/XDC uses to

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 SSecurity)**

represent zaps to load modules. Consistent with that change, the various security topics in the Online Help have been updated to describe the new rule formats, not the old ones.

So this **OLDRULES** topic has been added both to describe the old formats and to offer suggestions about converting existing rules from the old formats to the new ones.

### **HELP SECURITY GZAP**

Another major change is the final dropping of support for the old XDC.GZAP rule. So this topic has been added to discuss this issue and to offer suggestions for replacing an existing XDC.GZAP rule with a newer, supported rule(s).

### **HELP SECURITY CACHE**

For performance reasons, z/XDC maintains a local cache of security rulings. This topic discusses that cache and the security exposures created and not created by it.

### **HELP SECURITY TRACE**

A new feature in this release is the ability to produce a trace of calls that z/XDC makes to its system security interface. This topic describes in detail the information contained in that trace.

## **Help Whatsnew Z18 SRbmode**

A major enhancement for this release is the addition of support for debugging SRB mode routines. Just set up z/XDC as your SRB's newest FRR (Functional Recovery Routine). Do so in the same way that you would for your own FRR, but designate z/XDC's normal entry address in place of your own FRR routine.

Every effort has been taken to make SRB mode debugging as noninvasive to the SRB routine as possible. However, there is one **major problem**: z/XDC must release all locks held by the SRB routine whenever z/XDC receives control! (This is unavoidable.)

Whenever, z/XDC returns control to the SRB routine, it reacquires all prior locks before doing so; however, the consequences of this compromise are profound! For complete details, see **HELP DEBUGGING FRR**.

A new security rule (XDC.LOSTLOCKS) has been implemented that allows the control of who will and will not be permitted to resume the execution of a program after locks have been lost (as a result of passing control to z/XDC in the first place).

**WARNING!** Unlike all other XDC related security rules, XDC will interpret NOT PROTECTED for this rule as being **ACCESS DENIED**. Therefore, if you wish to use z/XDC to debug programs that hold locks, this XDC.LOSTLOCKS rule will have to be created, and you will need to be given READ access to it. For more information, see **HELP SECURITY LOSTLOCKS**.

Note, Whenever z/XDC receives control and is able to detect that locks have been lost, it issues message DBC933W to alert you to this circumstance.

For more information, see **HELP WHATSNEW Z18 FRR**.

# z/XDC® z1.8 RELEASE GUIDE

## Help Whatsnew Z18 SToragescrolling

In response to requests for the ability to "scroll through storage", I have made a change to the **DOWN** command to achieve a scroll like behavior. Whenever a **DOWN** command would cause the display window to move past the end of the session log, **and** if the last thing that had been displayed in the session log was a storage display, then z/XDC will now automatically issue a **DISPLAY** or **FORMAT** command (as appropriate) to append a display of the next chunk of storage to the bottom of the log, thereby extending the log and bringing that next chunk of storage into view. For more information, see HELP COMMANDS DOWN.

A similar change has **not** been made to the **UP** command, but if **UP** commands are issued after a series of scroll extending DOWN commands, then a limited upwards scrolling effect is achieved.

## Help Whatsnew Z18 Xdccall

The following changes and improvements have been made to the XDCCALL utility (XDCCALL, XDCCALLA, XDCCMD and XDCCMDA):

At the start of the debugging session, the PSW will now point to the target program's entry point instruction. (Previously, it had pointed to a BR R15 instruction that was about to jump to the program's entry point.) So now an initial **WHERE** command will display your program's entry point logic.

When your program first receives control (from XDCCALL at the start of the debugging session), R6 through R12 will now be zeroed. (R2-R5 will continue to contain non-zero text data, and R13-R1, of course, will continue to contain standard entry linkage data.)

XDCCALL now supports IBM's "standard" AMODE(64) entry linkage. In particular, **R15 will not** point to the user program's entry point. Instead (following IBM's documentation on this), R15 will contain X'FFFFFF02' indicating that the user program's caller (XDCCALL) is running in AMODE(31). For more information, refer to the **Passing Control with Return** topic in **Chapter. 4 Program Management** of the **z/OS MVS Programming: Assembler Services Guide** (SA22-7605).

Support has been added for two new keyword ddnames:

- **//xxxQUICK DD DUMMY** causes XDCCALL to pass control directly to the user program without first stopping at the beginning of the program. See HELP WHATSNEW Z18 QUICKSTART for more information.
- **//xxxRENT8 DD DUMMY** provides an alternative way by which XDCCALL can cause reentrant programs to be loaded into key 8 storage for problem state and key debugging. See HELP BREAKPOINTS REENTRANT for more information.

## Help Whatsnew Z18 Z9

Support has been added to z/XDC for recognition of machine instructions that are new with both the Z9-109 processor and the Z9-BC processor. These include:

AFI	CLGFI	LGHR	LT	MYH	SLGFI
AGFI	CSST	LHI	LTG	MYHR	STCKF
AGHI	CU21	LHR	MAY	MYL	STFLE

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 Z9)**

AHI	CU24	LLC	MAYH	MYLR	TROO
ALFI	ECTG	LLCR	MAYHR	MYR	TROT
ALGFI	FLOGR	LLGCR	MAYL	NIHF	TRTO
BRASL	IIHF	LLGHR	MAYLR	NILF	TRTT
CFI	IILF	LLH	MAYR	OIHF	XIHF
CGFI	LBR	LLHR	MGHI	OILF	XILF
CGHI	LGBR	LLIHF	MHI	PTFF	
CHI	LGFI	LLILF	MVCOS	SERV	
CLFI	LGHI	LPTEA	MY	SLFI	

## **Help Whatsnew Z18 Miscellaneous**

The following minor changes have been made to z/XDC.

### **Recommended Display Width**

The recommended display width has been increased from 130 characters to 136. This is in consideration of the fact that IPCS will generate storage displays that show 32 bytes of storage per line when the screen display width is 136 columns or wider. (For narrower displays, IPCS will show only 16 bytes of storage per line.) For more information, see HELP FULLSCREEN TERMINALS GEOMETRIES.

### **User Commands Exit**

The User Commands Exit Interface has been changed as follows:

- Both error level and retry level state data are now made available to the exit.
- The exit may now change retry level state data.
- The **SRCXUCMD** sample exit has been modified to look for an external routine named **USRXUCMD** and to call it if found. This permits the customer to segregate his local code out of the SRCXUCMD source code that we distribute, thereby insulating his code from changes that we might distribute in maintenance or new releases.
- The SRCXUCMD sample exit now can interface to user written z/XDC commands written in the REXX programming language. (See HELP WHATSNEW Z18 REXX for more information.)

For more information, see HELP EXITS USERCMDS.

### **Watch Windows**

Support for blank command lines has been restored. Now, if the user blanks out a watch window's command line, it will stay blanked out, and the window's contents will remain unchanged. (Previously, blank command lines were always restored to their prior non-blank contents.)

Normally, when a watch window's command line is non-blank, each press of the ENTER key (or of a PF-key) causes the commands to be executed and, therefore, the window contents updated. One customer made the suggestion that he would like to have a way

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 MIscellaneous)**

to stop the updating of a watch window's display and, thus, preserve a particular display for awhile. Allowing blank command lines accomplishes this objective.

For more information about watch windows, see HELP FULLSCREEN WINDOWS.

### **XDCCICSX**

z/XDC's **XDC** transaction's support has been improved for using XDC clones to debug CICS programs. The XDCCICSX program has been updated to look for a //XDCISxxx DD DUMMY allocation by which it can determine the name of the XDC clone to use when z/XDC's name has been changed from "XDC" to something else. For more information, see HELP DEBUGGING CICS.

### **#XDCHOOK and #DIE Macros**

The #XDCHOOK and #DIE macros are now sensitive to the ARCHLVL= operand of IBM's SYSSTATE macro:

- When ARCHLVL=1 (or greater) is set, #XDCHOOK and #DIE will generate jump instructions instead of branch instructions. In addition, the code generated by the #DIE macro will require no predefined base registers.
- When ARCHLVL=2 (or greater) is set, the code generated by the #XDCHOOK macro will require no predefined base registers.

For more information, see:

HELP BREAKPOINTS DEADTRAPS #DIE  
HELP DEBUGGING HOOKS STATIC #XDCHOOK

## **Help Whatsnew Z18 Incompatibilities**

Several changes have been made in this release that give rise to some incompatibilities and to certain special recommendations. For detailed information, you can select the following topics directly, or you can use HELP \*NEXT to proceed sequentially. Use HELP \*FORWARD to skip.

<b>DANELEN</b>	- This value has been increased.
<b>EXITS</b>	- Miscellaneous-exits interface change
<b>FIND</b>	- FIND command operand changes
<b>PFKEYS</b>	- Factory default PF-key settings changes
<b>POINTANDSHOOT</b>	- The way in which Point-and-Shoot commands work has been changed.
<b>RETRIEVE</b>	- RETRIEVE command operand changes
<b>SECURITY</b>	- Security rule changes
<b>SETBANG</b>	- Changed factory default setting for the exclamation point (!) operator
<b>SHOW</b>	- SHOW command operand changes
<b>XDCACIF</b>	- Changes to the #DBCXACP macro may require some recoding in your local XDCACIF exit routine (if any).
<b>XDCPANEL</b>	- Changes to z/XDC's Startup Panel in ISPF

## **Help Whatsnew Z18 Incompatibilities Danelen**

The value of the **DANELEN** equate has been increased from X'8000' to X'F000'.

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 Incompatibilities Danelen)**

This matters for those customers that have written a z/XDC Front/Backend routine for preallocating z/XDC's temporary work area. (If you don't know what DANELEN is, then don't worry about it.)

The required size of this work area has increased from 8 pages to 15 pages. For more information, see commentary located within the source code for the #DBCPRM macro (available in the XDCMACS library. The library's factory default name is DBCOLE.XDCZ18.XDCMACS. Ask your Systems Programmer for its actual name at your Data Center.)

## **Help Whatsnew Z18 Incompatibilities Exits**

The "Miscellaneous Exits" module is an optional router module used for a variety of exit calls from z/XDC. For more information about this, see HELP EXITS MISCXITS.

Because of a validity check requirement, code has been added to z/XDC to insure that the first machine instruction at the entry point for the Miscellaneous Exits module must be an R15-based unconditional branch, as follows:

```
      47F0 Fxxx      BC  15,xxx(,R15)
or:   47F0 Fxxx      B   xxx(,R15)
```

If the Miscellaneous Exits module's first instruction is not as shown above, then the module will be ignored. (This change was made to aid in the detection of gross corruption of an exit routine. One customer had a situation where his exits module was deleted from storage without z/XDC being notified. Things went south pretty badly when the module's storage was reused and z/XDC, not knowing this, tried to call the module.)

## **Help Whatsnew Z18 Incompatibilities Find**

If the FIND command should produce a formatted display of storage, then the **INSTRUCTION**, **DATA**, and **NOBIAS** operands now affect the display in the same way they would if used on a FORMAT command. (Previously, they only affected the attributes of equates produced via the EQUATENAMES= operand.) For more information, see HELP COMMANDS FIND.

## **Help Whatsnew Z18 Incompatibilities PFkeys**

In order to take advantage of new capabilities of the SET WINDOW CREATE and SET WINDOW DELETE commands, The factory default values for PF-keys 1 and 13 have been changed. A hyphen (-) has been added to the end of the command strings to allow the user to provide command operands from the command line. For more information, see:

```
HELP FULLSCREEN PFKEYS DFLTKEYS SETA
HELP FULLSCREEN PFKEYS DFLTKEYS SETBC
HELP COMMANDS SET WINDOW CREATEDELETE
```

A **NXSEQ(6)** term has been added to the TRACE command in the factory default value for PF-key 11. The value now is:

```
T NXSEQ() NXSEQ(2) NXSEQ(3) NXSEQ(4) NXSEQ(5) NXSEQ(6) ;GOT
```

For more information, see HELP FULLSCREEN PFKEYS DFLTKEYS SETA.

# *z/XDC® z1.8 RELEASE GUIDE*

## Help Whatsnew Z18 Incompatibilities POintandshoot

Changes have been made in the way in which Point-and-Shoot commands work. The width of the pointers used by such commands is no longer limited by the width of the hex data display field from which they are extracted. See HELP WHATSNEW Z18 POINTANDSHOOT for more information.

## Help Whatsnew Z18 Incompatibilities Retrieve

When the RETRIEVE command (or its aliases, RETRY and RECALL) is given with an **unsigned** command number operand, it is now treated as a negative number. This means that the nth **prior** command is retrieved. (Previously, it meant the nth following command was retrieved.)

The meaning of a **signed** command number remains unchanged: A negative command number causes a prior command to be retrieved. A signed positive number causes a following command to be retrieved. (Or a wrap to the oldest command occurs when the newest has already been retrieved.) For more information, see HELP COMMANDS RETRIEVE.

Also, due to parsing complications that are specific to RETRIEVE command processing, abbreviations of both the RETRIEVE command and its aliases (RETRY and RECALL) will no longer be accepted. Now, whenever these commands are used, they must be fully spelled out.

## Help Whatsnew Z18 Incompatibilities SECURITY

z/XDC's support for System Security Rules has changed in ways that create both current and future incompatibilities with existing Data Center security rule definitions! See HELP WHATSNEW Z18 SECURITY for all the gory details.

## Help Whatsnew Z18 Incompatibilities SETbang

The factory default setting for the exclamation point (!) operator has changed. Previously, it had been AMODE sensitive. Now, it just means **64-bit indirect**.

If you wish to use an AMODE-sensitive indirect operator in an address expression, then you can do either of two things:

- You can issue a **SET BANG AMODE** command, then use the exclamation point operator (!) as you've done in the past.
- Or you can use the **~INDIRECT (AMODE)** built-in function in any place where you would have used an AMODE-sensitive exclamation point. (Useful mainly when used in scripts.)

If you already have your own profile set up, then this change will not affect you. So if you want to propagate this change to your own profile, then issue the following sequence of commands:

```
PROFILE READ          (to insure that no unwanted changes creep in)
SET BANG 64BIT      (to cause "!" to mean 64-bit indirect)
```

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Help Whatsnew Z18 Incompatibilities SETbang)**

**PROFILE SAVE** (to save your updated profile)

### **Help Whatsnew Z18 Incompatibilities SHow**

The SHOW command no longer accepts the SOURCE, OBJECT, and BOTH operands. SHOW now unconditionally displays only formatted object code (SYM data), never source code (ADATA). For more information, see HELP COMMANDS SHOW.

### **Help Whatsnew Z18 Incompatibilities XDCAcif**

This is of interest only to those Customers who have a locally written XDCACIF exit (a routine that, when present, z/XDC calls instead of System Security for its various access control needs - see HELP EXITS ACIF for more information).

A change has been made to the **#DBCXACP** macro that affects the way in which certain fields are defined. Those fields that are a part of "extension sections" of the XACPL control block have been moved into separate dsects. So if your exit references those fields, then you will have to recode those references. See the source code for the **#DBCXACP** macro for details. That macro can be found in the **XDCMACS** distribution library. (That library's factory default name is DBCOLE.XDCZ18.XDCMACS.)

Also, two new security call types have been defined:

- **XRTREFRU** (=6) has been defined to assist in the conversion of certain load module related security rules (for controlling zapping) from an old format to a new format.
- **XRTLOCKS** (=7) has been defined to represent security requests controlling the resumption of the user's program when locks have been lost.

### **Help Whatsnew Z18 Incompatibilities XDCPanel**

A new **Quick Start?** field has been added to z/XDC's Startup Panel, so be sure to replace at least the following panel related elements when you upgrade to this release:

**XDCPANEL:** The Startup Panel itself.

**XDCPHLP3:** The ISPF help panel describing the Quick Start field.

**XDCCLIST:** A clist that interfaces between the Startup Panel and the debugging session.

**Notice:** These panel changes are **not** actually incompatible with older versions and releases of XDC. Older XDCs will, of course, ignore any settings the user might attempt to make in the new **Quick Start?** field; however, other than that, the new panel is completely compatible with older XDCs.

For more information, see HELP WHATSNEW Z18 QUICKSTART.

# ***z/XDC<sup>®</sup> z1.8 RELEASE GUIDE***

# z/XDC® z1.8 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) alphabets, and that only some special characters sort in front of alphabets. Others sort behind alphabets.

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.

! (indirect operator)	
default meaning changed1	<a href="#">28</a>
incompatibilities with prior releases?	<a href="#">28</a>
INDIRECT(...) built-in function9	<a href="#">19</a>
profiled default*	<a href="#">19</a>
SET BANG command*	<a href="#">19</a>
#DBCPARM macro	<a href="#">27</a>
#DBCXACP macro	
ACIF security exit routine-	<a href="#">29</a>
Incompatibilities with prior releases8	<a href="#">29</a>
XDCACIF security exit routine0	<a href="#">29</a>
XRTLOCKS field!	<a href="#">29</a>
XRTRETRU field!	<a href="#">29</a>
#DIE macro	
&SYSALVL global symbol%	<a href="#">26</a>
ARCHLVL setting	<a href="#">26</a>
base registers	<a href="#">26</a>
relative-immediate branching+	<a href="#">26</a>
SYSSTATE macro	<a href="#">26</a>
system architectural level)	<a href="#">26</a>
#XDCHOOK macro	
&SYSALVL global symbol)	<a href="#">26</a>
ARCHLVL setting"	<a href="#">26</a>
base registers!	<a href="#">26</a>
relative-immediate branching/	<a href="#">26</a>
SYSSTATE macro!	<a href="#">26</a>
system architectural level-	<a href="#">26</a>
&SYSALVL global symbol	
#DIE macro%	<a href="#">26</a>
#XDCHOOK macro)	<a href="#">26</a>
@ERn built-in equates	
storage displays*	<a href="#">14</a>
@ERWn built-in equate	<a href="#">13</a>
@ERWn built-in equates	
storage displays+	<a href="#">14</a>
@Rn built-in equates	
storage displays)	<a href="#">14</a>
@RWn built-in equate	<a href="#">13</a>
@RWn built-in equates	
storage displays*	<a href="#">14</a>
abends	
SDUMPX macro	<a href="#">13</a>
SDWAREC flag	<a href="#">13</a>
xxxSDUMP ddname	<a href="#">13</a>
ACIF security exit routine	
#DBCXACP macro-	<a href="#">29</a>

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

Incompatibilities with prior releasesD	29
XRTLOCKS field-	29
XRTREFRU field-	29
ADATA libraries	
xxxMAPLB ddname#	16
ADATA maps	
MCALL assembler option%	5
NOMCALL assembler option'	5
ADEFERRED command	
conditional traces(	6, 11
AFI machine instruction	25
AGFI machine instruction	25
AGHI machine instruction	25
AHI machine instruction	25
ALFI machine instruction	25
ALGFI machine instruction	25
ARCHLVL setting	
#DIE macro	26
#XDCHOOK macro"	26
artifacts of the debugging session	
'	5
LIST ESTAES commands;	5
LIST LSTACK commands;	6
LIST RBS commands8	5
LIST SCBS commands9	5
request blocks5	5, 6
assembler options	
MCALL	5, 14
NOMCALL	5
AT command	
conditional traces!	6, 11
ATX command	
conditional traces"	6, 11
base registers	
#DIE macro	26
#XDCHOOK macro!	26
BOTH operand	
SHOW command (not)#	11, 29
BRASL machine instruction	25
built-in equates	
@ERn	14
@ERWn	13, 14
@Rn	14
@RWn	13, 14
built-in functions	
INDIRECT(...)\$	19
CDF	
mixed case passwords support\$	6
CFI machine instruction	25
CGFI machine instruction	25
CGHI machine instruction	25
change history	
z/XDC z1.8	3
CHI machine instruction	25
CICS	

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

XDC clones	26
XDCCICSX transaction	26
XDCISxxx ddname	26
CLFI machine instruction	25
CLGFI machine instruction	25
command lines	
watch windows	25
command scripts	
SYSINFO	11
commands	
ADEFERRED	6, 11
AT	6, 11
ATX	6, 11
COMMENTARY	6
CONSOLE	7
CONSOLE command (deleted)&	12
DELETE MAPLIBS	7
DOWN	7, 14, 24
FIND	7
FORMAT	7
GO	7
GOT	7
GOX	7
LIST CONSOLE command!	12
LIST CONSOLES	8
LIST ESTAES	5, 8
LIST FEATURES	8
LIST FORMAT	8
LIST LOCKS	8, 15
LIST LSES	8
LIST LSTACK	6, 8
LIST RBS	5, 9
LIST SCBS	5, 8
LIST SECURITY	9
LIST STATISTICS	9
new (see new commands)#	6
point-and-shoot commands%	17
R	9
READ	9
RETRIEVE	9
RETRIEVE LIST	9
REXX (see XRI)	9
REXX commands (see XRI command),	11, 20
SET BANG	10, 28
SET CONSOLE	10
SET CONSOLE command	12
SET FORMAT	10, 19
SET LOCKS	10, 15
SET MAPLIBS	10
SET SECURITY	10
SET WINDOW CREATE	10
SET WINDOW DELETE	10
SHOW	11
TDEFERRED	6, 11
TRACE	6, 11

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

TRAP	6, 11
UP	24
WHERE	11
XRI	11, 20
COMMENTARY command	6
DOWN command#	6
DOWN HALF command(	6
UP command!	6
conditional traces	
ADEFERRED command(	6, 11
AT command!	6, 11
ATX command"	6, 11
compound conditions*	6
nesting	6
TDEFERRED command(	6, 11
TRACE command\$	6, 11
TRAP command#	6, 11
CONSOLE command	
deleted	7
CONSOLE command (deleted)	12
consoles	
4-byte console IDs	12
CONSOLE command (deleted)&	12
console names	12
EMCS consoles	12
LIST CONSOLE command!	12
MCS consoles	12
message routing	11
MODIFY xxxCDF command response routing3	12
SET CONSOLE command	12
START xxxCDF command response routing2	12
STOP xxxCDF command response routing1	12
user interface	12
CONSOLES operand	
LIST command!	8
contacts	
Online Help for	17
COPY assembler statement	
LIST FORMAT command0	8
COPYd code	
LIST FORMAT command"	8
credits	
Online Help for	17
Cross Domain Facility (see CDF)	
\$	6
CSST machine instruction	25
CU21 machine instruction	25
CU24 machine instruction	25
CURRENTMCODE operand	
FIND command%	5, 7, 16
FORMAT command'	5, 7, 16
macro expansions)	14, 15
profiled setting)	19
SET FORMAT command+	5, 10, 16
WHERE command&	5, 11, 16

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

DANELEN equate	<a href="#">27</a>
DATA operand	
FIND command	<a href="#">7, 27</a>
DBC933W message	
lost locks	<a href="#">15, 22, 23</a>
DBCnnn messages	
displaying via the SHOW command3	<a href="#">11</a>
DBCPARM control block	
DANELEN equate(	<a href="#">27</a>
ddnames	
XDCISxxx ddname	<a href="#">26</a>
xxxMAPLB	<a href="#">16</a>
xxxQUICK	<a href="#">20</a>
xxxSDUMP	<a href="#">13</a>
xxxTRACE	<a href="#">16</a>
xxxTRSAF	<a href="#">22</a>
xxxXRI	<a href="#">20</a>
debugging session	
artifacts	<a href="#">5</a>
DELETE command	
MAPLIBS operand"	<a href="#">7</a>
DELETE MAPLIBS command	<a href="#">7</a>
QUIET operand(	<a href="#">7</a>
deleted commands	
CONSOLE	<a href="#">7</a>
DISPLAY command	
formatting Consistency*	<a href="#">14</a>
issued by the DOWN command.	<a href="#">14, 24</a>
display geometries	
62x136	<a href="#">25</a>
DISPLAY= operand	
FIND command!	<a href="#">7</a>
DOWN command	
COMMENTARY command#	<a href="#">6</a>
issuing a DISPLAY command*	<a href="#">14, 24</a>
issuing a FORMAT command)	<a href="#">14, 24</a>
scrolling through storage*	<a href="#">7, 14, 24</a>
DOWN HALF command	
COMMENTARY command(	<a href="#">6</a>
dumps	
Online Help for	<a href="#">17</a>
e-mail (see internet)	<a href="#">ii</a>
ECTG machine instruction	<a href="#">25</a>
EMCS consoles	<a href="#">12</a>
equates	
built-in (see built-in equates)+	<a href="#">13</a>
register equates (see register equates)3	<a href="#">14</a>
error level registers	
@ERWn built-in equates0	<a href="#">13</a>
error level state data	
user commands exit routine5	<a href="#">25</a>
ESTAES operand	
LIST command	<a href="#">8</a>
exit routines	
user commands	<a href="#">25</a>

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

FEATURES operand	
LIST command!	<u>8</u>
fields	
SDWAREC	<u>13</u>
FIND command	
CURRENTMCODE operand%	<u>5, 7, 16</u>
DATA operand	<u>7, 27</u>
DISPLAY= operand!	<u>7</u>
FORMAT= operand	<u>7</u>
HIDEMCODE operand"	<u>5, 7, 16</u>
incompatibilities with prior releases6	<u>27</u>
INSTRUCTION operand\$	<u>7, 27</u>
macro expansions!	<u>7, 16</u>
NOBIAS operand	<u>7, 27</u>
processing statistics&	<u>7</u>
SHOWMCODE operand"	<u>5, 7, 16</u>
STATISTICS operand#	<u>7</u>
FLOGR machine instruction	<u>25</u>
FORMAT command	<u>7</u>
CURRENTMCODE operand'	<u>5, 7, 16</u>
formatting Consistency)	<u>14</u>
HIDEMCODE operand\$	<u>5, 7, 16</u>
issued by the DOWN command-	<u>14, 24</u>
macro expansions#	<u>7, 16</u>
SHOWMCODE operand\$	<u>5, 7, 16</u>
FORMAT operand	
LIST command	<u>8</u>
FORMAT= operand	
FIND command	<u>7</u>
formatting Consistency	<u>13</u>
DISPLAY command*	<u>14</u>
FORMAT command)	<u>14</u>
format control options1	<u>13</u>
line commands(	<u>13</u>
point and shoot commands3	<u>14</u>
scrolling through storage4	<u>14</u>
FRR routines	
LIST LOCKS command#	<u>15</u>
lost locks	<u>15</u>
running x/XDC as!	<u>15</u>
SET LOCKS command"	<u>15</u>
XDC.LOSTLOCKS security rule,	<u>15</u>
FTP address (see internet)	<u>ii</u>
Functional Recovery Routine (see FRR)	
*	<u>15</u>
functions (see built-in functions)	
'	<u>19</u>
GO command	<u>7</u>
GOT command	<u>7</u>
GOX command	<u>7</u>
HALF operand	
COMMENTARY command#	<u>6</u>
DOWN command	<u>6</u>
HIDEMCODE operand	
FIND comand!	<u>16</u>

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

FIND command"	<a href="#">5, 7</a>
FORMAT comand#	<a href="#">16</a>
FORMAT command\$	<a href="#">5, 7</a>
macro expansions&	<a href="#">14, 15</a>
profiled setting&	<a href="#">19</a>
SET FORMAT comand'	<a href="#">16</a>
SET FORMAT command(	<a href="#">5, 10</a>
WHERE comand"	<a href="#">16</a>
WHERE command#	<a href="#">5, 11</a>
history	
z/XDC z1.8	<a href="#">3</a>
home page (see internet)	<a href="#">ii</a>
IIHF machine instruction	<a href="#">25</a>
IILF machine instruction	<a href="#">25</a>
incompatibilities	
RETRIEVE command&	<a href="#">28</a>
incompatibilities with prior releases	
! (indirect operator)?	<a href="#">28</a>
#DBCXACP macro8	<a href="#">29</a>
ACIF security exit routineD	<a href="#">29</a>
FIND command6	<a href="#">27</a>
Miscellaneous Exits Router routineL	<a href="#">27</a>
PF-keys1	<a href="#">27</a>
RETRIEVE command:	<a href="#">28</a>
security rules8	<a href="#">28</a>
SET BANG command:	<a href="#">28</a>
SHOW command6	<a href="#">29</a>
XDC.GZAP security rule@	<a href="#">20</a>
XDC.ZAP.COMMON.MODULE.modulename rulesP	<a href="#">21</a>
XDC.ZAP.COMMON.modulename rulesI	<a href="#">21</a>
XDC.ZAP.PRIVATE.MODULE.modulename rulesQ	<a href="#">21</a>
XDC.ZAP.PRIVATE.modulename rulesJ	<a href="#">21</a>
XDCACIF security exit routineG	<a href="#">29</a>
indirect operators	
!	<a href="#">28</a>
exclamation point (!),	<a href="#">28</a>
INDIRECT(...) built-in function	
! (indirect operator)9	<a href="#">19</a>
SET BANG command4	<a href="#">19</a>
INSTRUCTION operand	
FIND command\$	<a href="#">7, 27</a>
internet	
e-mail address	<a href="#">ii</a>
FTP address	<a href="#">ii</a>
home page	<a href="#">ii, iii</a>
web address	<a href="#">ii</a>
ISPF5XX script	
Online Help for"	<a href="#">17</a>
LBR machine instruction	<a href="#">25</a>
legal statements	
trademark notice%	<a href="#">iii</a>
usage warning"	<a href="#">ii</a>
LGBR machine instruction	<a href="#">25</a>
LGFI machine instruction	<a href="#">25</a>
LGHI machine instruction	<a href="#">25</a>

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

LGHR machine instruction	<u>25</u>
LHI machine instruction	<u>25</u>
LHR machine instruction	<u>25</u>
libraries	
XDCMACS macro library#	<u>27</u>
line commands	
formatting Consistency(	<u>13</u>
LIST command	
ESTAE operand renamed to ESTAES0	<u>8</u>
ESTAES operand	<u>8</u>
FEATURES operand!	<u>8</u>
FORMAT operand	<u>8</u>
LOCKS operand	<u>8</u>
LSES operand	<u>8</u>
LSTACK operand	<u>8</u>
RBS operand	<u>9</u>
SCBS operand	<u>8</u>
SECURITY operand!	<u>9</u>
LIST CONSOLE command	
consoles!	<u>12</u>
LIST CONSOLES command	<u>8</u>
LIST ESTAE command	
renamed to LIST ESTAES-	<u>8</u>
LIST ESTAES command	<u>8</u>
LIST ESTAES commands	
artifacts of the debugging session;	<u>5</u>
LIST FEATURES command	<u>8</u>
LPTEA machine instruction3	<u>8</u>
mixed case passwords support6	<u>8</u>
STCKF machine instruction3	<u>8</u>
STFLE machine instruction3	<u>8</u>
LIST FORMAT command	<u>8</u>
COPY assembler statement0	<u>8</u>
COPYd code"	<u>8</u>
macro expansions(	<u>8</u>
SET LINES value'	<u>8</u>
LIST LOCKS command	<u>8</u>
FRR routines#	<u>15</u>
lost locks!	<u>8, 15</u>
security rules%	<u>8, 15</u>
LIST LSES command	<u>8</u>
LIST LSTACK command	<u>8</u>
LIST LSTACK commands	
artifacts of the debugging session;	<u>6</u>
LIST RBS command	<u>9</u>
LIST RBS commands	
artifacts of the debugging session8	<u>5</u>
LIST SCBS command	<u>8</u>
LIST SCBS commands	
artifacts of the debugging session9	<u>5</u>
LIST SECURITY command	<u>9</u>
LIST STATISTICS command	<u>9</u>
LLC machine instruction	<u>25</u>
LLCR machine instruction	<u>25</u>
LLGCR machine instruction	<u>25</u>

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

LLGHR machine instruction	<a href="#">25</a>
LLH machine instruction	<a href="#">25</a>
LLHR machine instruction	<a href="#">25</a>
LLIHF machine instruction	<a href="#">25</a>
LLILF machine instruction	<a href="#">25</a>
load modules	
security rule changes&	<a href="#">21</a>
locks (lost)	
DBC933W message	<a href="#">15</a> , <a href="#">22</a> , <a href="#">23</a>
FRR routines	<a href="#">15</a>
LIST LOCKS command#	<a href="#">8</a> , <a href="#">15</a>
SET LOCKS command"	<a href="#">15</a>
XDC.LOSTLOCKS security rule,	<a href="#">22</a>
LOCKS operand	
LIST command	<a href="#">8</a>
LPTEA machine instruction	<a href="#">25</a>
LIST FEATURES command3	<a href="#">8</a>
LSES operand	
LIST command	<a href="#">8</a>
LSTACK operand	
LIST command	<a href="#">8</a>
LT machine instruction	<a href="#">25</a>
LTG machine instruction	<a href="#">25</a>
machine instructions	
AFI	<a href="#">25</a>
AGFI	<a href="#">25</a>
AGHI	<a href="#">25</a>
AHI	<a href="#">25</a>
ALFI	<a href="#">25</a>
ALGFI	<a href="#">25</a>
BRASL	<a href="#">25</a>
CFI	<a href="#">25</a>
CGFI	<a href="#">25</a>
CGHI	<a href="#">25</a>
CHI	<a href="#">25</a>
CLFI	<a href="#">25</a>
CLGFI	<a href="#">25</a>
CSST	<a href="#">25</a>
CU21	<a href="#">25</a>
CU24	<a href="#">25</a>
ECTG	<a href="#">25</a>
FLOGR	<a href="#">25</a>
IIHF	<a href="#">25</a>
IILF	<a href="#">25</a>
LBR	<a href="#">25</a>
LGBR	<a href="#">25</a>
LGFI	<a href="#">25</a>
LGHI	<a href="#">25</a>
LGHR	<a href="#">25</a>
LHI	<a href="#">25</a>
LHR	<a href="#">25</a>
LLC	<a href="#">25</a>
LLCR	<a href="#">25</a>
LLGCR	<a href="#">25</a>
LLGHR	<a href="#">25</a>

# **z/XDC<sup>®</sup> z1.8 RELEASE GUIDE**

## **(Index)**

LLH	25
LLHR	25
LLIHF	25
LLILF	25
LPTEA	8, 25
LT	25
LTG	25
MAY	25
MAYH	25
MAYHR	25
MAYL	25
MAYLR	25
MAYR	25
MGHI	25
MHI	25
MVCOS	25
MY	25
MYH	25
MYHR	25
MYL	25
MYLR	25
MYR	25
NIHF	25
NILF	25
OIHF	25
OILF	25
PTFF	25
SERVC	25
SLFI	25
SLGFI	25
STCKF	8, 25
STFLE	8, 25
TROO	25
TROT	25
TRTO	25
TRTT	25
XIHF	25
XILF	25
Z9-109 processor)	24
Z9-BC processor(	24
macro expansions	
CURRENTMCODE operand)	14, 15
FIND command!	7
FORMAT command#	7
HIDEMCODE operand&	14, 15
LIST FORMAT command(	8
MCALL assembler option+	14
SET FORMAT command'	10
SHOWMCODE operand&	14, 15
suppression	14, 15
WHERE command"	11
macros	
#DBCPARM	27
#DIE	26
#XDCHOOK	26

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

SDUMPX .....	<a href="#">13</a>
MAPLIBS operand	
DELETE command" .....	<a href="#">7</a>
MAY machine instruction .....	<a href="#">25</a>
MAYH machine instruction .....	<a href="#">25</a>
MAYHR machine instruction .....	<a href="#">25</a>
MAYL machine instruction .....	<a href="#">25</a>
MAYLR machine instruction .....	<a href="#">25</a>
MAYR machine instruction .....	<a href="#">25</a>
MCALL assembler option	
ADATA maps% .....	<a href="#">5</a>
macro expansion displays3 .....	<a href="#">14</a>
MCS consoles .....	<a href="#">12</a>
messages	
DBC933W .....	<a href="#">15</a> , <a href="#">22</a> , <a href="#">23</a>
MGHI machine instruction .....	<a href="#">25</a>
MHI machine instruction .....	<a href="#">25</a>
Miscellaneous Exits Router routine	
incompatibility with prior releasesJ .....	<a href="#">27</a>
mixed case passwords support	
CDF\$ .....	<a href="#">6</a>
LIST FEATURES command6 .....	<a href="#">8</a>
MODIFY xxxCDF operator command	
respose routine2 .....	<a href="#">12</a>
MVCOS machine instruction .....	<a href="#">25</a>
MY machine instruction .....	<a href="#">25</a>
MYH machine instruction .....	<a href="#">25</a>
MYHR machine instruction .....	<a href="#">25</a>
MYL machine instruction .....	<a href="#">25</a>
MYLR machine instruction .....	<a href="#">25</a>
MYR machine instruction .....	<a href="#">25</a>
new commands .....	<a href="#">6</a>
LIST LOCKS .....	<a href="#">8</a>
LIST SECURITY .....	<a href="#">9</a>
SET BANG .....	<a href="#">10</a>
SET LOCKS .....	<a href="#">10</a>
SET SECURITY .....	<a href="#">10</a>
SYSINFO command script' .....	<a href="#">11</a>
NIHF machine instruction .....	<a href="#">25</a>
NILF machine instruction .....	<a href="#">25</a>
nn operand	
SET WINDOW CREATE command( .....	<a href="#">10</a>
SET WINDOW DELETE command( .....	<a href="#">10</a>
NOBIAS operand	
FIND command .....	<a href="#">7</a> , <a href="#">27</a>
NOMCALL assembler option	
ADATA maps' .....	<a href="#">5</a>
NXSEQ(...) built-in function	
PF-key 11 factory default: .....	<a href="#">27</a>
OBJECT operand	
SHOW command (not)% .....	<a href="#">11</a> , <a href="#">29</a>
OIHF machine instruction .....	<a href="#">25</a>
OILF machine instruction .....	<a href="#">25</a>
Online Help	
contacts .....	<a href="#">17</a>

# z/XDC® z1.8 RELEASE GUIDE

## (Index)

credits	17
dumps	17
ISPF5XX script	17
profiles	16, 18
XDC clones	17
operands	
cmd#	9
CONSOLES	8
CURRENTMCODE	7, 10, 11, 14, 15, 19
DATA	7, 27
DISPLAY=	7
ESTAES	8
FEATURES	8
FORMAT	8
FORMAT=	7
HALF	6
HIDEMCODE	7, 10, 11, 14, 15, 19
INSTRUCTION	7, 27
LIST	9
LOCKS	8
LSES	8
LSTACK	8
MAPLIBS	7
nn	10
NOBIAS	7, 27
QUIET	7, 10
RBS	9
SCBS	8
SECURITY	9
SHOEMCODE	19
SHOWMCODE	7, 10, 11, 14, 15
STATISTICS	7
operator commands	
MODIFY xxxCDF#	12
START xxxCDF"	12
STOP xxxCDF!	12
operator consoles (see consoles)	
%	11
operators	
!(indirect operator)#	28
PF-key 01 factory default setting	
SET WINDOW DELETE -9	10, 27
PF-key 11 factory default setting	
T ... NXSEQ(6);GOT8	27
PF-key 13 factory default setting	
SET WINDOW CREATE -9	10, 27
PF-keys	
incompatibilities with prior releases1	27
PF-key 01 factory default: SET WINDOW DELETE -:	10, 27
PF-key 03 factory default: SET WINDOW CREATE -:	10
PF-key 11 factory default: T ... NXSEQ(6);GOT9	27
PF-key 13 factory default: SET WINDOW CREATE -:	27
point and shoot commands	
formatting Consistency3	14
point-and-shoot commands	

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

width limit changes0	17
profiles	
changes to the factory defaults,	18
CURRENTMCODE setting!	19
FIND command display default change0	19
HELP PF-key 216 changes\$	19
HELP PF-key 96 changes#	19
HIDEMCODE setting	19
making changes safely"	19
Online Help for	16, 18
PF-key 1 changes	18
PF-key 11 changes	18
PF-key 13 changes	18
PF-key 21 changes	18
PF-key 25 changes	18
PF-key 26 changes	19
PF-key 33 changes	18
RETRIEVE list changes"	19
SET BANG default change\$	19
SET FORMAT command	19
SHOWMCODE setting	19
programs	
XDCCALL	20
PTFF machine instruction	25
quick start	
XDCCALL program	20
XDCLIST clist	29
XDCPANEL panel	20, 29
XDCPHLP3 panel	29
xxxQUICK ddname	20
QUIET operand	
DELETE MAPLIBS command(	7
SET MAPLIBS command%	10
R command	
abbreviation for READ command+	9
RBS operand	
LIST command	9
READ command	
Minimum abbreviation: R(	9
RECALL command (see RETRIEVE command)	
*	9
register equates	
@ERWn	13
@RWn	13
storage displays%	14
RETRIEVE command	
abbreviations"	28
cmd# operand!	9
incompatibilities with prior releases:	28
incompatibilities&	28
LIST operand!	9
unsigned cmd# operand*	28
RETRY command (see RETRIEVE command)	
)	9
retry level registers	

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

@RWn built-in equates/	<a href="#">13</a>
retry level state data	
user commands exit routine5	<a href="#">25</a>
REXX command (see XRI command)	
#	<a href="#">9</a>
REXX commands (see XRI command)	
\$	<a href="#">11, 20</a>
SCBS operand	
LIST command	<a href="#">8</a>
scripts	
ISPF5XX	<a href="#">17</a>
scripts (see command scripts)	
"	<a href="#">11</a>
scrolling through storage	
DISPLAY command-	<a href="#">14, 24</a>
DOWN comma	<a href="#">24</a>
DOWN command*	<a href="#">7, 14</a>
FORMAT command,	<a href="#">14, 24</a>
formatting Consistency4	<a href="#">14</a>
UP command(	<a href="#">24</a>
SDUMPX macro	
xxxSDUMP ddname	<a href="#">13</a>
SDWAREC flag	
unexpected abends"	<a href="#">13</a>
xxxSDUMP ddname	<a href="#">13</a>
SECURITY operand	
LIST command!	<a href="#">9</a>
security rules	
FRR routines	<a href="#">15</a>
incompatibilities with prior releases8	<a href="#">28</a>
LIST LOCKS command%	<a href="#">8, 15</a>
SRB routines	<a href="#">23</a>
XDC.GZAP rule	<a href="#">20</a>
XDC.LOSTLOCKS	<a href="#">22</a>
XDC.REFEREE.ZAPMODULERULES-	<a href="#">21</a>
XDC.ZAP.COMMON.MODULE.modulename rules9	<a href="#">21</a>
XDC.ZAP.COMMON.modulename rules2	<a href="#">21</a>
XDC.ZAP.PRIVATE.MODULE.modulename rules:	<a href="#">21</a>
XDC.ZAP.PRIVATE.modulename rules3	<a href="#">21</a>
zapping load modules, rule chaznges6	<a href="#">21</a>
security trace displays	
SET SECURITY command0	<a href="#">22</a>
xxxTRSAF ddname+	<a href="#">22</a>
SERV machine instruction	<a href="#">25</a>
Service Request Block (see SRB)	
\$	<a href="#">23</a>
SET BANG command	<a href="#">10</a>
! (indirect operator)*	<a href="#">19</a>
default setting changed,	<a href="#">28</a>
profiled default%	<a href="#">19</a>
SET CONSOLE command	
consoles	<a href="#">12</a>
SET FORMAT command	
CURRENTMCODE operand+	<a href="#">5, 10, 16</a>
HIDEMCODE operand(	<a href="#">5, 10, 16</a>

# **z/XDC® z1.8 RELEASE GUIDE**

## **(Index)**

macro expansions'	<a href="#">10, 16</a>
profiled settings(	<a href="#">19</a>
SHOWMCODE operand(	<a href="#">5, 10, 16</a>
SET LOCKS command	<a href="#">10</a>
FRR routines"	<a href="#">15</a>
lost locks	<a href="#">15</a>
SET MAPLIBS command	
QUIET operand%	<a href="#">10</a>
SET SECURITY command	<a href="#">10</a>
security trace displays0	<a href="#">22</a>
SET WINDOW CREATE command	
nn operand(	<a href="#">10</a>
PF-key 13 factory default7	<a href="#">10, 27</a>
SET WINDOW DELETE command	
nn operand(	<a href="#">10</a>
PF-key 01 factory default7	<a href="#">10, 27</a>
SHOW command	
BOTH operand (not)#	<a href="#">11, 29</a>
displaying DBCnnn messages+	<a href="#">11</a>
OBJECT operand (not)%	<a href="#">11, 29</a>
SOURCE operand (not)%	<a href="#">11, 29</a>
SHOWMCODE operand	
FIND comand!	<a href="#">16</a>
FIND command"	<a href="#">5, 7</a>
FORMAT comand#	<a href="#">16</a>
FORMAT command\$	<a href="#">5, 7</a>
macro expansions&	<a href="#">14, 15</a>
profiled setting&	<a href="#">19</a>
SET FORMAT comand'	<a href="#">16</a>
SET FORMAT command(	<a href="#">5, 10</a>
WHERE comand"	<a href="#">16</a>
WHERE command#	<a href="#">5, 11</a>
SLFI machine instruction	<a href="#">25</a>
SLGFI machine instruction	<a href="#">25</a>
source modules	
SRCXUCMD (user commands exit)0	<a href="#">25</a>
SOURCE operand	
SHOW command (not)%	<a href="#">11, 29</a>
SRB	
debugging	<a href="#">23</a>
SRB routines	
XDC.LOSTLOCKS security rule,	<a href="#">23</a>
SRCXUCMD source module	<a href="#">25</a>
USRXUCMD user module/	<a href="#">25</a>
START xxxCDF operator command	
respose routine1	<a href="#">12</a>
STATISTICS operand	
FIND command#	<a href="#">7</a>
STCKF machine instruction	<a href="#">25</a>
LIST FEATURES command3	<a href="#">8</a>
STFLE machine instruction	<a href="#">25</a>
LIST FEATURES command3	<a href="#">8</a>
STOP xxxCDF operator command	
respose routine0	<a href="#">12</a>
storage displays	

# **z/XDC<sup>®</sup> z1.8 RELEASE GUIDE**

## **(Index)**

register equates%	<a href="#">14</a>
SYSINFO command script	<a href="#">11</a>
SYSSTATE macro	
#DIE macro	<a href="#">26</a>
#XDCHOOK macro!	<a href="#">26</a>
system operator commands (see operator commands)	
5	<a href="#">12</a>
system operator consoles (see consoles)	
,	<a href="#">11</a>
TDEFERRED command	
conditional traces(	<a href="#">6</a> , <a href="#">11</a>
TRACE command	
conditional traces\$	<a href="#">6</a> , <a href="#">11</a>
PF-key 11 factory default+	<a href="#">27</a>
trademarks notice	<a href="#">iii</a>
TRAP command	
conditional traces#	<a href="#">6</a> , <a href="#">11</a>
TROO machine instruction	<a href="#">25</a>
TROT machine instruction	<a href="#">25</a>
TRTO machine instruction	<a href="#">25</a>
TRTT machine instruction	<a href="#">25</a>
unexpected abends	
SDUMPX macro"	<a href="#">13</a>
SDWAREC flag"	<a href="#">13</a>
xxxSDUMP ddname%	<a href="#">13</a>
UP command	
COMMENTARY command!	<a href="#">6</a>
scrolling through storage(	<a href="#">24</a>
usage warning	<a href="#">ii</a>
user commands exit routine	
error level state data5	<a href="#">25</a>
retry level state data5	<a href="#">25</a>
SRCXUCMD sourc module4	<a href="#">25</a>
user interface	
consoles	<a href="#">12</a>
system operator consoles+	<a href="#">12</a>
USRXUCMD user module	
SRCXUCMD source module/	<a href="#">25</a>
watch windows	
command lines	<a href="#">25</a>
web address (see internet)	<a href="#">ii</a>
WHERE command	
CURRENTMCODE operand&	<a href="#">5</a> , <a href="#">11</a> , <a href="#">16</a>
HIDEMCODE operand#	<a href="#">5</a> , <a href="#">11</a> , <a href="#">16</a>
macro expansions"	<a href="#">11</a> , <a href="#">16</a>
SHOWMCODE operand#	<a href="#">5</a> , <a href="#">11</a> , <a href="#">16</a>
WTO/WTOs	
console routine	<a href="#">11</a>
XDC clones	
CICS	<a href="#">26</a>
Online Help for	<a href="#">17</a>
XDCCICSX transaction#	<a href="#">26</a>
XDCISxxx ddname	<a href="#">26</a>
XDC.GZAP security rule	
Incompatibilities with prior releases@	<a href="#">20</a>

# z/XDC® z1.8 RELEASE GUIDE

## (Index)

no longer supported. . . . .	<a href="#">20</a>
XDC.LOSTLOCKS security rule	
default is DENIED!2 . . . . .	<a href="#">15, 22, 23</a>
FRR routines, . . . . .	<a href="#">15</a>
locks (lost), . . . . .	<a href="#">22</a>
security rule changes5 . . . . .	<a href="#">22</a>
SRB routines, . . . . .	<a href="#">23</a>
XDC.REFEREE.ZAPMODULERULES security rule	
security rule changesB . . . . .	<a href="#">21</a>
XDC.ZAP.COMMON.MODULE.modulename security rule	
security rule changesH . . . . .	<a href="#">21</a>
XDC.ZAP.COMMON.modulename security rule	
security rule changesA . . . . .	<a href="#">21</a>
XDC.ZAP.PRIVATE.MODULE.modulename security rule	
security rule changesI . . . . .	<a href="#">21</a>
XDC.ZAP.PRIVATE.modulename security rule	
security rule changesB . . . . .	<a href="#">21</a>
XDC/REXX interface (see XRI command)	
) . . . . .	<a href="#">11</a>
XDCACIF security exit routine	
#DBCXACP macro0 . . . . .	<a href="#">29</a>
Incompatibilities with prior releasesG . . . . .	<a href="#">29</a>
XDCCALL program	
quick start . . . . .	<a href="#">20</a>
XDCCICSX transaction	
XDC clones# . . . . .	<a href="#">26</a>
XDCISxxx ddname( . . . . .	<a href="#">26</a>
XDCCLIST clist	
quick start . . . . .	<a href="#">29</a>
XDCISxxx ddname	
CICS . . . . .	<a href="#">26</a>
XDC clones . . . . .	<a href="#">26</a>
XDCCICSX transaction( . . . . .	<a href="#">26</a>
XDCMACS macro library	
#DBCPARM macro( . . . . .	<a href="#">27</a>
XDCPANEL panel	
quick start . . . . .	<a href="#">20, 29</a>
XDCPHLP3 panel	
quick start . . . . .	<a href="#">29</a>
XDCXRI ddname (see xxxXRI ddname)	
& . . . . .	<a href="#">20</a>
XIHF machine instruction . . . . .	<a href="#">25</a>
XILF machine instruction . . . . .	<a href="#">25</a>
XRI command	
REXX commands . . . . .	<a href="#">11, 20</a>
xxxXRI ddname . . . . .	<a href="#">20</a>
XRTLOCKS field	
#DBCXACP macro! . . . . .	<a href="#">29</a>
ACIF security exit routine- . . . . .	<a href="#">29</a>
XRTREFRU field	
#DBCXACP macro! . . . . .	<a href="#">29</a>
ACIF security exit routine- . . . . .	<a href="#">29</a>
xxxMAPLB ddname	
ADATA libraries# . . . . .	<a href="#">16</a>
xxxQUICK ddname	

# **z/XDC<sup>®</sup> z1.8 RELEASE GUIDE**

## **(Index)**

quick start .....	<a href="#">20</a>
xxxSDUMP ddname	
SDUMPX macro .....	<a href="#">13</a>
SDWAREC flag .....	<a href="#">13</a>
unexpected abends% .....	<a href="#">13</a>
xxxTRACE ddname	
terminal communications problems4 .....	<a href="#">16</a>
xxxTRSAF ddname	
security trace displays+ .....	<a href="#">22</a>
xxxXRI ddname	
XRI command .....	<a href="#">20</a>
Z9-109 processor	
new machine instructions- .....	<a href="#">24</a>
Z9-BC processor	
new machine instructions, .....	<a href="#">24</a>
zapping load modules	
security rule changes. ....	<a href="#">21</a>

# ***z/XDC<sup>®</sup> z1.8 RELEASE GUIDE***

# ***z/XDC<sup>®</sup> z1.8 RELEASE GUIDE***