

---

# DECwindows Motif for OpenVMS Guide to Non-C Bindings

Order Number: AA-PGZ8B-TE

**January 1994**

This guide supplies non-C bindings for programmers developing DECwindows Motif for OpenVMS applications in languages other than C. Pascal and Fortran are supported. Programmers can choose from among OSF/Motif and ancillary widgets supplied by Digital.

|                                     |                                                                                                    |
|-------------------------------------|----------------------------------------------------------------------------------------------------|
| <b>Revision/Update Information:</b> | This is a revised manual.                                                                          |
| <b>Operating System:</b>            | OpenVMS AXP Version 1.5<br>VMS Version 5.5-2                                                       |
| <b>Software Version:</b>            | DECwindows Motif Version 1.2<br>for OpenVMS AXP<br>DECwindows Motif Version 1.2<br>for OpenVMS VAX |

**Digital Equipment Corporation  
Maynard, Massachusetts**

---

**January 1994**

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

© Digital Equipment Corporation 1994. All Rights Reserved.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation: Alpha AXP, AXP, Bookreader, DECwindows, Digital, OpenVMS, VAX, VAX DOCUMENT, VMS, and the DIGITAL logo.

The following are third-party trademarks:

Open Software Foundation is a trademark, and OSF/Motif and Motif are registered trademarks of the Open Software Foundation, Inc.

X Window System is a trademark of the Massachusetts Institute of Technology.

All other trademarks and registered trademarks are the property of their respective holders.

ZK5636

This document is available on CD-ROM.

This document was prepared using DECdocument, Version 3.3-1b.

---

# Contents

|                                                                                    |      |
|------------------------------------------------------------------------------------|------|
| <b>Preface</b> .....                                                               | xxxv |
| <b>1 Introduction</b>                                                              |      |
| 1.1 Xlib Routine Template .....                                                    | 1-1  |
| 1.2 Intrinsic, OSF/Motif Toolkit, and Digital Extension Routines<br>Template ..... | 1-5  |
| <b>Part I DECwindows Motif for OpenVMS Xlib</b>                                    |      |
| <b>2 Xlib Routines</b>                                                             |      |
| ACTIVATE SCREEN SAVER .....                                                        | 2-1  |
| ADD HOST .....                                                                     | 2-1  |
| ADD HOSTS .....                                                                    | 2-2  |
| ADD PIXEL .....                                                                    | 2-2  |
| ADD TO SAVE SET .....                                                              | 2-2  |
| ALLOC CLASS HINT .....                                                             | 2-3  |
| ALLOC COLOR .....                                                                  | 2-3  |
| ALLOC COLOR CELLS .....                                                            | 2-4  |
| ALLOC COLOR PLANES .....                                                           | 2-5  |
| ALLOC ICON SIZE .....                                                              | 2-5  |
| ALLOC NAMED COLOR .....                                                            | 2-6  |
| ALLOC SIZE HINTS .....                                                             | 2-6  |
| ALLOC STANDARD COLORMAP .....                                                      | 2-7  |
| ALLOW EVENTS .....                                                                 | 2-7  |
| ALL PLANES .....                                                                   | 2-8  |
| AUTO REPEAT OFF .....                                                              | 2-8  |
| AUTO REPEAT ON .....                                                               | 2-8  |
| BELL .....                                                                         | 2-9  |
| BITMAP BIT ORDER .....                                                             | 2-9  |
| BITMAP PAD .....                                                                   | 2-10 |
| BITMAP UNIT .....                                                                  | 2-10 |
| BLACK PIXEL .....                                                                  | 2-10 |
| BLACK PIXEL OF SCREEN .....                                                        | 2-11 |
| CELLS OF SCREEN .....                                                              | 2-11 |
| CHANGE ACTIVE POINTER GRAB .....                                                   | 2-11 |
| CHANGE GC .....                                                                    | 2-12 |
| CHANGE KEYBOARD CONTROL .....                                                      | 2-13 |

|                                      |      |
|--------------------------------------|------|
| CHANGE KEYBOARD MAPPING .....        | 2-13 |
| CHANGE POINTER CONTROL .....         | 2-14 |
| CHANGE PROPERTY .....                | 2-14 |
| CHANGE SAVE SET .....                | 2-15 |
| CHANGE WINDOW ATTRIBUTES .....       | 2-16 |
| CHECK IF EVENT .....                 | 2-16 |
| CHECK MASK EVENT .....               | 2-16 |
| CHECK TYPED EVENT .....              | 2-17 |
| CHECK TYPED WINDOW EVENT .....       | 2-17 |
| CHECK WINDOW EVENT .....             | 2-18 |
| CIRCULATE SUBWINDOWS .....           | 2-18 |
| CIRCULATE SUBWINDOWS DOWN .....      | 2-19 |
| CIRCULATE SUBWINDOWS UP .....        | 2-19 |
| CLEAR AREA .....                     | 2-19 |
| CLEAR WINDOW .....                   | 2-20 |
| CLIP BOX .....                       | 2-20 |
| CLOSE DISPLAY .....                  | 2-21 |
| CONFIGURE WINDOW .....               | 2-21 |
| CONNECTION NUMBER .....              | 2-22 |
| CONVERT SELECTION .....              | 2-22 |
| COPY AREA .....                      | 2-23 |
| COPY COLORMAP AND FREE .....         | 2-23 |
| COPY GC .....                        | 2-24 |
| COPY PLANE .....                     | 2-24 |
| CREATE BITMAP FROM DATA .....        | 2-25 |
| CREATE COLORMAP .....                | 2-25 |
| CREATE FONT CURSOR .....             | 2-26 |
| CREATE GC .....                      | 2-26 |
| CREATE GLYPH CURSOR .....            | 2-27 |
| CREATE IMAGE .....                   | 2-27 |
| CREATE PIXMAP .....                  | 2-28 |
| CREATE PIXMAP FROM BITMAP DATA ..... | 2-29 |
| CREATE REGION .....                  | 2-29 |
| CREATE SIMPLE WINDOW .....           | 2-29 |
| CREATE WINDOW .....                  | 2-30 |
| DEFAULT COLORMAP .....               | 2-31 |
| DEFAULT COLORMAP OF SCREEN .....     | 2-31 |
| DEFAULT DEPTH .....                  | 2-31 |
| DEFAULT DEPTH OF SCREEN .....        | 2-32 |
| DEFAULT GC .....                     | 2-32 |
| DEFAULT GC OF SCREEN .....           | 2-32 |
| DEFAULT ROOT WINDOW .....            | 2-33 |
| DEFAULT SCREEN .....                 | 2-33 |
| DEFAULT SCREEN OF DISPLAY .....      | 2-34 |
| DEFAULT VISUAL .....                 | 2-34 |
| DEFAULT VISUAL OF SCREEN .....       | 2-35 |
| DEFINE CURSOR .....                  | 2-35 |

|                                  |      |
|----------------------------------|------|
| DELETE CONTEXT .....             | 2-35 |
| DELETE MODIFIERMAP ENTRY .....   | 2-36 |
| DELETE PROPERTY .....            | 2-37 |
| DESTROY IMAGE .....              | 2-37 |
| DESTROY REGION .....             | 2-37 |
| DESTROY SUBWINDOWS .....         | 2-38 |
| DESTROY WINDOW .....             | 2-38 |
| DISABLE ACCESS CONTROL .....     | 2-38 |
| DISPLAY CELLS .....              | 2-39 |
| DISPLAY HEIGHT .....             | 2-39 |
| DISPLAY HEIGHT MM .....          | 2-40 |
| DISPLAY KEYCODES .....           | 2-40 |
| DISPLAY MOTION BUFFER SIZE ..... | 2-40 |
| DISPLAY NAME .....               | 2-41 |
| DISPLAY OF SCREEN .....          | 2-42 |
| DISPLAY PLANES .....             | 2-42 |
| DISPLAY STRING .....             | 2-42 |
| DISPLAY WIDTH .....              | 2-43 |
| DISPLAY WIDTH MM .....           | 2-44 |
| DOES BACKING STORE .....         | 2-44 |
| DOES SAVE UNDERS .....           | 2-45 |
| DRAW ARC .....                   | 2-45 |
| DRAW ARCS .....                  | 2-46 |
| DRAW IMAGE STRING .....          | 2-46 |
| DRAW IMAGE STRING 16 .....       | 2-47 |
| DRAW LINE .....                  | 2-47 |
| DRAW LINES .....                 | 2-48 |
| DRAW POINT .....                 | 2-49 |
| DRAW POINTS .....                | 2-49 |
| DRAW RECTANGLE .....             | 2-50 |
| DRAW RECTANGLES .....            | 2-50 |
| DRAW SEGMENTS .....              | 2-51 |
| DRAW STRING .....                | 2-51 |
| DRAW STRING 16 .....             | 2-52 |
| DRAW TEXT .....                  | 2-52 |
| DRAW TEXT 16 .....               | 2-53 |
| EMPTY REGION .....               | 2-54 |
| ENABLE ACCESS CONTROL .....      | 2-54 |
| EQUAL REGION .....               | 2-54 |
| EVENT MASK OF SCREEN .....       | 2-55 |
| EVENTS QUEUED .....              | 2-55 |
| FETCH BUFFER .....               | 2-56 |
| FETCH BYTES .....                | 2-56 |
| FETCH NAME .....                 | 2-57 |
| FILL ARC .....                   | 2-58 |
| FILL ARCS .....                  | 2-58 |
| FILL POLYGON .....               | 2-59 |

|                               |      |
|-------------------------------|------|
| FILL RECTANGLE .....          | 2-60 |
| FILL RECTANGLES .....         | 2-60 |
| FILTER EVENT .....            | 2-61 |
| FIND CONTEXT .....            | 2-61 |
| FLUSH .....                   | 2-62 |
| FLUSH GC .....                | 2-62 |
| FORCE SCREEN SAVER .....      | 2-62 |
| FREE .....                    | 2-63 |
| FREE COLORMAP .....           | 2-63 |
| FREE COLORS .....             | 2-64 |
| FREE CURSOR .....             | 2-64 |
| FREE FONT .....               | 2-65 |
| FREE GC .....                 | 2-65 |
| FREE MODIFIERMAP .....        | 2-65 |
| FREE PIXMAP .....             | 2-66 |
| FREE STRING LIST .....        | 2-66 |
| GCONTEXT FROM GC .....        | 2-66 |
| GEOMETRY .....                | 2-67 |
| GET ATOM NAME .....           | 2-68 |
| GET CHAR STRUCT .....         | 2-69 |
| GET CLASS HINT .....          | 2-70 |
| GET COMMAND .....             | 2-70 |
| GET DEFAULT .....             | 2-71 |
| GET ERROR DATABASE TEXT ..... | 2-72 |
| GET ERROR TEXT .....          | 2-73 |
| GET FONT PATH .....           | 2-73 |
| GET FONT PROPERTY .....       | 2-74 |
| GET GC VALUES .....           | 2-75 |
| GET GEOMETRY .....            | 2-75 |
| GET ICON NAME .....           | 2-77 |
| GET ICON SIZES .....          | 2-78 |
| GET IMAGE .....               | 2-79 |
| GET INPUT FOCUS .....         | 2-80 |
| GET KEYBOARD CONTROL .....    | 2-80 |
| GET KEYBOARD MAPPING .....    | 2-81 |
| GET MODIFIER MAPPING .....    | 2-82 |
| GET MOTION EVENTS .....       | 2-82 |
| GET NORMAL HINTS .....        | 2-83 |
| GET PIXEL .....               | 2-84 |
| GET POINTER CONTROL .....     | 2-84 |
| GET POINTER MAPPING .....     | 2-85 |
| GET RGB COLORMAPS .....       | 2-86 |
| GET SCREEN SAVER .....        | 2-87 |
| GET SELECTION OWNER .....     | 2-88 |
| GET SIZE HINTS .....          | 2-88 |
| GET STANDARD COLORMAP .....   | 2-88 |
| GET SUB IMAGE .....           | 2-89 |

|                                        |       |
|----------------------------------------|-------|
| GET TEXT PROPERTY . . . . .            | 2-90  |
| GET TRANSIENT FOR HINT . . . . .       | 2-91  |
| GET VISUAL INFO . . . . .              | 2-91  |
| GET WINDOW ATTRIBUTES . . . . .        | 2-92  |
| GET WINDOW PROPERTY . . . . .          | 2-93  |
| GET WM CLIENT MACHINE . . . . .        | 2-94  |
| GET WM COLORMAP WINDOWS . . . . .      | 2-95  |
| GET WM HINTS . . . . .                 | 2-96  |
| GET WM ICON NAME . . . . .             | 2-96  |
| GET WM NAME . . . . .                  | 2-97  |
| GET WM NORMAL HINTS . . . . .          | 2-97  |
| GET WM PROTOCOLS . . . . .             | 2-98  |
| GET ZOOM HINTS . . . . .               | 2-99  |
| GRAB BUTTON . . . . .                  | 2-100 |
| GRAB KEY . . . . .                     | 2-102 |
| GRAB KEYBOARD . . . . .                | 2-103 |
| GRAB POINTER . . . . .                 | 2-104 |
| GRAB SERVER . . . . .                  | 2-106 |
| HEIGHT MM OF SCREEN . . . . .          | 2-106 |
| HEIGHT OF SCREEN . . . . .             | 2-106 |
| ICONIFY WINDOW . . . . .               | 2-107 |
| IF EVENT . . . . .                     | 2-107 |
| IMAGE BYTE ORDER . . . . .             | 2-108 |
| INSERT MODIFIERMAP ENTRY . . . . .     | 2-108 |
| INSTALL COLORMAP . . . . .             | 2-109 |
| INTERN ATOM . . . . .                  | 2-110 |
| INTERSECT REGION . . . . .             | 2-110 |
| KEYCODE TO KEYSYM . . . . .            | 2-110 |
| KEYSYM TO KEYCODE . . . . .            | 2-111 |
| KEYSYM TO STRING . . . . .             | 2-111 |
| KILL CLIENT . . . . .                  | 2-112 |
| LAST KNOWN REQUEST PROCESSED . . . . . | 2-112 |
| LIST DEPTHS . . . . .                  | 2-113 |
| LIST FONT . . . . .                    | 2-113 |
| LIST FONTS . . . . .                   | 2-114 |
| LIST FONT WITH INFO . . . . .          | 2-114 |
| LIST FONTS WITH INFO . . . . .         | 2-116 |
| LIST HOSTS . . . . .                   | 2-117 |
| LIST INSTALLED COLORMAPS . . . . .     | 2-118 |
| LIST PIXMAP FORMATS . . . . .          | 2-119 |
| LIST PROPERTIES . . . . .              | 2-120 |
| LOAD FONT . . . . .                    | 2-121 |
| LOAD QUERY FONT . . . . .              | 2-121 |
| LOOKUP COLOR . . . . .                 | 2-122 |
| LOOKUP KEYSYM . . . . .                | 2-123 |
| LOOKUP STRING . . . . .                | 2-123 |
| LOWER WINDOW . . . . .                 | 2-124 |

|                             |       |
|-----------------------------|-------|
| MAP RAISED .....            | 2-124 |
| MAP SUBWINDOWS .....        | 2-124 |
| MAP WINDOW .....            | 2-125 |
| MASK EVENT .....            | 2-125 |
| MATCH VISUAL INFO .....     | 2-125 |
| MAX CMAPS OF SCREEN .....   | 2-126 |
| MAX REQUEST SIZE .....      | 2-126 |
| MIN CMAPS OF SCREEN .....   | 2-127 |
| MOVE RESIZE WINDOW .....    | 2-127 |
| MOVE WINDOW .....           | 2-127 |
| NEW MODIFIERMAP .....       | 2-128 |
| NEXT EVENT .....            | 2-129 |
| NEXT REQUEST .....          | 2-129 |
| NO OP .....                 | 2-129 |
| OFFSET REGION .....         | 2-130 |
| OPEN DISPLAY .....          | 2-130 |
| PARSE COLOR .....           | 2-131 |
| PARSE GEOMETRY .....        | 2-132 |
| PEEK EVENT .....            | 2-133 |
| PEEK IF EVENT .....         | 2-133 |
| PENDING .....               | 2-134 |
| PERM ALLOC .....            | 2-134 |
| PLANES OF SCREEN .....      | 2-134 |
| POINT IN REGION .....       | 2-135 |
| POLYGON REGION .....        | 2-135 |
| PROTOCOL REVISION .....     | 2-136 |
| PROTOCOL VERSION .....      | 2-136 |
| PUT BACK EVENT .....        | 2-137 |
| PUT IMAGE .....             | 2-137 |
| PUT PIXEL .....             | 2-138 |
| Q LENGTH .....              | 2-138 |
| QUERY BEST CURSOR .....     | 2-138 |
| QUERY BEST SIZE .....       | 2-139 |
| QUERY BEST STIPPLE .....    | 2-140 |
| QUERY BEST TILE .....       | 2-141 |
| QUERY COLOR .....           | 2-141 |
| QUERY COLORS .....          | 2-142 |
| QUERY FONT .....            | 2-142 |
| QUERY KEYPAD .....          | 2-143 |
| QUERY POINTER .....         | 2-143 |
| QUERY TEXT EXTENTS .....    | 2-144 |
| QUERY TEXT EXTENTS 16 ..... | 2-145 |
| QUERY TREE .....            | 2-146 |
| RAISE WINDOW .....          | 2-147 |
| READ BITMAP FILE .....      | 2-147 |
| REBIND KEYSYM .....         | 2-149 |
| RECOLOR CURSOR .....        | 2-149 |



|                                    |       |
|------------------------------------|-------|
| RECONFIGURE WM WINDOW .....        | 2-150 |
| RECT IN REGION .....               | 2-150 |
| REFRESH KEYBOARD MAPPING .....     | 2-151 |
| REMOVE FROM SAVE SET .....         | 2-151 |
| REMOVE HOST .....                  | 2-152 |
| REMOVE HOSTS .....                 | 2-152 |
| RESPARENT WINDOW .....             | 2-152 |
| RESET SCREEN SAVER .....           | 2-153 |
| RESIZE WINDOW .....                | 2-153 |
| RESOURCE MANAGER STRING .....      | 2-154 |
| RESTACK WINDOWS .....              | 2-154 |
| RM COMBINE DATABASE .....          | 2-155 |
| RM COMBINE FILE DATABASE .....     | 2-155 |
| RM DESTROY DATABASE .....          | 2-155 |
| RM ENUMERATE DATABASE .....        | 2-156 |
| RM GET DATABASE .....              | 2-156 |
| RM GET FILE DATABASE .....         | 2-157 |
| RM GET RESOURCE .....              | 2-157 |
| RM GET STRING DATABASE .....       | 2-158 |
| RM INITIALIZE .....                | 2-159 |
| RM LOCALE OF DATABASE .....        | 2-159 |
| RM MERGE DATABASES .....           | 2-159 |
| RM PARSE COMMAND .....             | 2-160 |
| RM PERM STRING TO QUARK .....      | 2-160 |
| RM PUT FILE DATABASE .....         | 2-161 |
| RM PUT LINE RESOURCE .....         | 2-161 |
| RM PUT RESOURCE .....              | 2-161 |
| RM PUT STRING RESOURCE .....       | 2-162 |
| RM Q GET RESOURCE .....            | 2-162 |
| RM Q GET SEARCH LIST .....         | 2-164 |
| RM Q GET SEARCH RESOURCE .....     | 2-164 |
| RM Q PUT RESOURCE .....            | 2-165 |
| RM Q PUT STRING RESOURCE .....     | 2-165 |
| RM QUARK TO STRING .....           | 2-166 |
| RM SET DATABASE .....              | 2-166 |
| RM STRING TO BIND QUARK LIST ..... | 2-167 |
| RM STRING TO QUARK .....           | 2-167 |
| RM STRING TO QUARK LIST .....      | 2-167 |
| RM UNIQUE QUARK .....              | 2-168 |
| ROOT WINDOW .....                  | 2-168 |
| ROOT WINDOW OF SCREEN .....        | 2-169 |
| ROTATE BUFFERS .....               | 2-169 |
| ROTATE WINDOW PROPERTIES .....     | 2-169 |
| SAVE CONTEXT .....                 | 2-170 |
| SCREEN COUNT .....                 | 2-170 |
| SCREEN NUMBER OF SCREEN .....      | 2-171 |
| SCREEN OF DISPLAY .....            | 2-171 |

|                               |       |
|-------------------------------|-------|
| SCREEN RESOURCE STRING .....  | 2-171 |
| SELECT ASYNC EVENT .....      | 2-172 |
| SELECT ASYNC INPUT .....      | 2-172 |
| SELECT INPUT .....            | 2-173 |
| SEND EVENT .....              | 2-173 |
| SERVER VENDOR .....           | 2-174 |
| SET ACCESS CONTROL .....      | 2-174 |
| SET AFTER FUNCTION .....      | 2-175 |
| SET ARC MODE .....            | 2-175 |
| SET BACKGROUND .....          | 2-176 |
| SET CLASS HINT .....          | 2-176 |
| SET CLIP MASK .....           | 2-177 |
| SET CLIP ORIGIN .....         | 2-177 |
| SET CLIP RECTANGLES .....     | 2-178 |
| SET CLOSE DOWN MODE .....     | 2-178 |
| SET COMMAND .....             | 2-179 |
| SET DASHES .....              | 2-179 |
| SET ERROR HANDLER .....       | 2-180 |
| SET FILL RULE .....           | 2-180 |
| SET FILL STYLE .....          | 2-181 |
| SET FONT .....                | 2-182 |
| SET FONT PATH .....           | 2-182 |
| SET FOREGROUND .....          | 2-182 |
| SET FUNCTION .....            | 2-183 |
| SET GRAPHICS EXPOSURES .....  | 2-184 |
| SET ICON NAME .....           | 2-184 |
| SET ICON SIZES .....          | 2-185 |
| SET INPUT FOCUS .....         | 2-185 |
| SET IO ERROR HANDLER .....    | 2-186 |
| SET LINE ATTRIBUTES .....     | 2-186 |
| SET MODIFIER MAPPING .....    | 2-187 |
| SET NORMAL HINTS .....        | 2-188 |
| SET PLANE MASK .....          | 2-188 |
| SET POINTER MAPPING .....     | 2-189 |
| SET REGION .....              | 2-189 |
| SET RGB COLORMAPS .....       | 2-190 |
| SET SCREEN SAVER .....        | 2-190 |
| SET SELECTION OWNER .....     | 2-191 |
| SET SIZE HINTS .....          | 2-191 |
| SET STANDARD COLORMAP .....   | 2-192 |
| SET STANDARD PROPERTIES ..... | 2-192 |
| SET STATE .....               | 2-193 |
| SET STIPPLE .....             | 2-194 |
| SET SUBWINDOW MODE .....      | 2-194 |
| SET TEXT PROPERTY .....       | 2-195 |
| SET TILE .....                | 2-195 |
| SET TRANSIENT FOR HINT .....  | 2-195 |

|                                    |       |
|------------------------------------|-------|
| SET TS ORIGIN .....                | 2-196 |
| SET WINDOW BACKGROUND .....        | 2-196 |
| SET WINDOW BACKGROUND PIXMAP ..... | 2-197 |
| SET WINDOW BORDER .....            | 2-197 |
| SET WINDOW BORDER PIXMAP .....     | 2-197 |
| SET WINDOW BORDER WIDTH .....      | 2-198 |
| SET WINDOW COLORMAP .....          | 2-198 |
| SET WM CLIENT MACHINE .....        | 2-199 |
| SET WM COLORMAP WINDOWS .....      | 2-199 |
| SET WM HINTS .....                 | 2-200 |
| SET WM ICON NAME .....             | 2-200 |
| SET WM NAME .....                  | 2-200 |
| SET WM NORMAL HINTS .....          | 2-201 |
| SET WM PROPERTIES .....            | 2-201 |
| SET WM PROTOCOLS .....             | 2-202 |
| SET WM SIZE HINTS .....            | 2-202 |
| SET ZOOM HINTS .....               | 2-203 |
| SHRINK REGION .....                | 2-203 |
| STORE BUFFER .....                 | 2-203 |
| STORE BYTES .....                  | 2-204 |
| STORE COLOR .....                  | 2-204 |
| STORE COLORS .....                 | 2-205 |
| STORE NAME .....                   | 2-206 |
| STORE NAMED COLOR .....            | 2-206 |
| STRING LIST TO TEXT PROPERTY ..... | 2-207 |
| STRING TO KEYSYM .....             | 2-207 |
| SUB IMAGE .....                    | 2-208 |
| SUBTRACT REGION .....              | 2-208 |
| SYNC .....                         | 2-208 |
| SYNCHRONIZE .....                  | 2-209 |
| TEXT EXTENTS .....                 | 2-209 |
| TEXT EXTENTS 16 .....              | 2-210 |
| TEXT PROPERTY TO STRING LIST ..... | 2-211 |
| TEXT WIDTH .....                   | 2-212 |
| TEXT WIDTH 16 .....                | 2-212 |
| TRANSLATE COORDINATES .....        | 2-212 |
| UNDEFINE CURSOR .....              | 2-213 |
| UNGRAB BUTTON .....                | 2-214 |
| UNGRAB KEY .....                   | 2-215 |
| UNGRAB KEYBOARD .....              | 2-215 |
| UNGRAB POINTER .....               | 2-216 |
| UNGRAB SERVER .....                | 2-216 |
| UNINSTALL COLORMAP .....           | 2-216 |
| UNION RECT WITH REGION .....       | 2-217 |
| UNION REGION .....                 | 2-217 |
| UNIQUE CONTEXT .....               | 2-218 |
| UNLOAD FONT .....                  | 2-218 |

|                             |       |
|-----------------------------|-------|
| UNMAP SUBWINDOWS .....      | 2-218 |
| UNMAP WINDOW .....          | 2-219 |
| VENDOR RELEASE .....        | 2-219 |
| VISUAL ID FROM VISUAL ..... | 2-219 |
| WARP POINTER .....          | 2-220 |
| WHITE PIXEL .....           | 2-220 |
| WHITE PIXEL OF SCREEN ..... | 2-221 |
| WIDTH MM OF SCREEN .....    | 2-221 |
| WIDTH OF SCREEN .....       | 2-221 |
| WINDOW EVENT .....          | 2-222 |
| WITHDRAW WINDOW .....       | 2-222 |
| WM GEOMETRY .....           | 2-223 |
| WRITE BITMAP FILE .....     | 2-223 |
| XOR REGION .....            | 2-224 |

### 3 Xlib Data Structures and Error Codes

|                                                   |      |
|---------------------------------------------------|------|
| 3.1 Any Event Data Structure .....                | 3-1  |
| 3.2 Arc Data Structure .....                      | 3-2  |
| 3.3 Button Event Data Structure .....             | 3-3  |
| 3.4 Char 2B Data Structure .....                  | 3-4  |
| 3.5 Char Struct Data Structure .....              | 3-5  |
| 3.6 Circulate Event Data Structure .....          | 3-5  |
| 3.7 Circulate Request Event Data Structure .....  | 3-6  |
| 3.8 Class Hint Data Structure .....               | 3-7  |
| 3.9 Client Message Event Data Structure .....     | 3-7  |
| 3.10 Color Data Structure .....                   | 3-8  |
| 3.11 Color Map Event Data Structure .....         | 3-9  |
| 3.12 Compose Status Data Structure .....          | 3-10 |
| 3.13 Configure Event Data Structure .....         | 3-11 |
| 3.14 Configure Request Event Data Structure ..... | 3-12 |
| 3.15 Create Window Event Data Structure .....     | 3-13 |
| 3.16 Crossing Event Data Structure .....          | 3-15 |
| 3.17 Destroy Window Event Data Structure .....    | 3-17 |
| 3.18 Error Event Data Structure .....             | 3-17 |
| 3.19 Event Data Structure .....                   | 3-18 |
| 3.19.1 The Event Mask .....                       | 3-19 |
| 3.20 Expose Event Data Structure .....            | 3-19 |
| 3.21 Ext Codes Data Structure .....               | 3-21 |
| 3.22 Ext Data Data Structure .....                | 3-21 |
| 3.23 Focus Change Event Data Structure .....      | 3-22 |
| 3.24 Font Property Data Structure .....           | 3-23 |
| 3.25 Font Data Structure .....                    | 3-24 |
| 3.26 The GC Values Data Structure .....           | 3-25 |
| 3.26.1 GC Mask .....                              | 3-28 |
| 3.27 Graphics Expose Event Data Structure .....   | 3-29 |
| 3.28 Gravity Event Data Structure .....           | 3-30 |
| 3.29 Host Address Data Structure .....            | 3-31 |
| 3.30 Icon Size Data Structure .....               | 3-32 |
| 3.31 Image Data Structure .....                   | 3-33 |
| 3.32 Key Event Data Structure .....               | 3-35 |
| 3.33 Keyboard Control Data Structure .....        | 3-36 |

|      |                                                  |      |
|------|--------------------------------------------------|------|
| 3.34 | Keyboard State Data Structure . . . . .          | 3-37 |
| 3.35 | Keymap Event Data Structure . . . . .            | 3-38 |
| 3.36 | Map Event Data Structure . . . . .               | 3-39 |
| 3.37 | Map Request Event Data Structure . . . . .       | 3-39 |
| 3.38 | Mapping Event Data Structure . . . . .           | 3-40 |
| 3.39 | Modifier Keymap Data Structure . . . . .         | 3-41 |
| 3.40 | Motion Event Data Structure . . . . .            | 3-42 |
| 3.41 | No Expose Event Data Structure . . . . .         | 3-43 |
| 3.42 | Pixmap Format Values Data Structure . . . . .    | 3-44 |
| 3.43 | Point Data Structure . . . . .                   | 3-45 |
| 3.44 | Property Event Data Structure . . . . .          | 3-45 |
| 3.45 | Rectangle Data Structure . . . . .               | 3-46 |
| 3.46 | Reparent Event Data Structure . . . . .          | 3-47 |
| 3.47 | Resize Request Event Data Structure . . . . .    | 3-48 |
| 3.48 | Resource Manager Option Data Structure . . . . . | 3-49 |
| 3.49 | Resource Manager Value Data Structure . . . . .  | 3-49 |
| 3.50 | Segment Data Structure . . . . .                 | 3-50 |
| 3.51 | Selection Clear Event Data Structure . . . . .   | 3-50 |
| 3.52 | Selection Event Data Structure . . . . .         | 3-51 |
| 3.53 | Selection Request Event Data Structure . . . . . | 3-52 |
| 3.54 | Set Window Attributes Data Structure . . . . .   | 3-53 |
| 3.55 | Size Hints Data Structure . . . . .              | 3-55 |
| 3.56 | Standard Color Map Data Structure . . . . .      | 3-56 |
| 3.57 | Text Item Data Structure . . . . .               | 3-57 |
| 3.58 | Text Item 16 Data Structure . . . . .            | 3-58 |
| 3.59 | Text Property Data Structure . . . . .           | 3-59 |
| 3.60 | Time Coordinate Data Structure . . . . .         | 3-59 |
| 3.61 | Unmap Event Data Structure . . . . .             | 3-60 |
| 3.62 | Visibility Event Data Structure . . . . .        | 3-61 |
| 3.63 | Visual Info Data Structure . . . . .             | 3-62 |
| 3.64 | Window Attributes Data Structure . . . . .       | 3-63 |
| 3.65 | Window Changes Data Structure . . . . .          | 3-66 |
| 3.66 | WM Hints Data Structure . . . . .                | 3-66 |
| 3.67 | Error Handling . . . . .                         | 3-67 |

## Part II OpenVMS DECwindows Intrinsics

### 4 Intrinsics Routines

|                                 |     |
|---------------------------------|-----|
| XtAddActions . . . . .          | 4-1 |
| XtAddCallback . . . . .         | 4-1 |
| XtAddCallbacks . . . . .        | 4-2 |
| XtAddConverter . . . . .        | 4-2 |
| XtAddEventHandler . . . . .     | 4-2 |
| XtAddExposureToRegion . . . . . | 4-3 |
| XtAddGrab . . . . .             | 4-3 |
| XtAddInput . . . . .            | 4-4 |
| XtAddRawEventHandler . . . . .  | 4-4 |
| XtAddTimeout . . . . .          | 4-5 |
| XtAddWorkProc . . . . .         | 4-5 |
| XtAllocateGC . . . . .          | 4-5 |
| XtAppAddActionHook . . . . .    | 4-6 |

|                               |      |
|-------------------------------|------|
| XtAppAddActions               | 4-6  |
| XtAppAddConverter             | 4-7  |
| XtAppAddInput                 | 4-7  |
| XtAppAddTimeOut               | 4-8  |
| XtAppAddWorkProc              | 4-8  |
| XtAppCreateShell              | 4-8  |
| XtAppError                    | 4-9  |
| XtAppErrorMsg                 | 4-9  |
| XtAppGetErrorDatabase         | 4-10 |
| XtAppGetErrorDatabaseText     | 4-10 |
| XtAppGetSelectionTimeout      | 4-11 |
| XtAppInitialize               | 4-11 |
| XtAppMainLoop                 | 4-12 |
| XtAppNextEvent                | 4-12 |
| XtAppPeekEvent                | 4-12 |
| XtAppPending                  | 4-13 |
| XtAppProcessEvent             | 4-13 |
| XtAppReleaseCacheRefs         | 4-13 |
| XtAppSetErrorHandler          | 4-14 |
| XtAppSetErrorMsgHandler       | 4-14 |
| XtAppSetFallbackResources     | 4-14 |
| XtAppSetSelectionTimeout      | 4-15 |
| XtAppSetTypeConverter         | 4-15 |
| XtAppSetWarningHandler        | 4-16 |
| XtAppSetWarningMsgHandler     | 4-16 |
| XtAppWarning                  | 4-16 |
| XtAppWarningMsg               | 4-17 |
| XtAugmentTranslations         | 4-17 |
| XtBuildEventMask              | 4-17 |
| XtCallAcceptFocus             | 4-18 |
| XtCallActionProc              | 4-18 |
| XtCallbackExclusive           | 4-19 |
| XtCallbackNone                | 4-19 |
| XtCallbackNonexclusive        | 4-19 |
| XtCallbackPopdown             | 4-20 |
| XtCallbackReleaseCacheRef     | 4-20 |
| XtCallbackReleaseCacheRefList | 4-21 |
| XtCallCallbackList            | 4-21 |
| XtCallCallbacks               | 4-21 |
| XtCallConverter               | 4-22 |
| XtCalloc                      | 4-22 |
| XtClass                       | 4-23 |
| XtCloseDisplay                | 4-23 |
| XtConvert                     | 4-23 |
| XtConvertAndStore             | 4-24 |
| XtConvertCase                 | 4-24 |
| XtCreateApplicationContext    | 4-25 |

|                               |      |
|-------------------------------|------|
| XtCreateApplicationShell      | 4-25 |
| XtCreateManagedWidget         | 4-25 |
| XtCreatePopupShell            | 4-26 |
| XtCreateWidget                | 4-26 |
| XtCvtColorToPixel             | 4-27 |
| XtCvtIntToBool                | 4-27 |
| XtCvtIntToBoolean             | 4-28 |
| XtCvtIntToColor               | 4-28 |
| XtCvtIntToFloat               | 4-29 |
| XtCvtIntToFont                | 4-29 |
| XtCvtIntToPixel               | 4-30 |
| XtCvtIntToPixmap              | 4-30 |
| XtCvtIntToShort               | 4-31 |
| XtCvtIntToUnsignedChar        | 4-31 |
| XtCvtStringToAcceleratorTable | 4-32 |
| XtCvtStringToAtom             | 4-32 |
| XtCvtStringToBool             | 4-33 |
| XtCvtStringToBoolean          | 4-33 |
| XtCvtStringToCursor           | 4-34 |
| XtCvtStringToDimension        | 4-34 |
| XtCvtStringToDisplay          | 4-35 |
| XtCvtStringToFile             | 4-35 |
| XtCvtStringToFloat            | 4-36 |
| XtCvtStringToFont             | 4-36 |
| XtCvtStringToFontSet          | 4-37 |
| XtCvtStringToFontStruct       | 4-37 |
| XtCvtStringToInt              | 4-38 |
| XtCvtStringToInitialState     | 4-38 |
| XtCvtStringToPixel            | 4-39 |
| XtCvtStringToShort            | 4-39 |
| XtCvtStringToTranslationTable | 4-40 |
| XtCvtStringToUnsignedChar     | 4-40 |
| XtCvtStringToVisual           | 4-41 |
| XtDatabase                    | 4-41 |
| XtDestroyApplicationContext   | 4-41 |
| XtDestroyGC                   | 4-42 |
| XtDestroyWidget               | 4-42 |
| XtDirectConvert               | 4-43 |
| XtDisownSelection             | 4-43 |
| XtDispatchEvent               | 4-43 |
| XtDisplay                     | 4-44 |
| XtDisplayInitialize           | 4-44 |
| XtDisplayOfObject             | 4-45 |
| XtDisplayStringConvWarning    | 4-45 |
| XtDisplayToApplicationContext | 4-45 |
| XtError                       | 4-46 |
| XtErrorMsg                    | 4-46 |

|                                 |      |
|---------------------------------|------|
| XtFindFile                      | 4-46 |
| XtFree                          | 4-47 |
| XtGetActionList                 | 4-47 |
| XtGetActionKeysym               | 4-48 |
| XtGetApplicationNameAndClass    | 4-48 |
| XtGetApplicationResources       | 4-48 |
| XtGetConstraintResourceList     | 4-49 |
| XtGetErrorDatabase              | 4-49 |
| XtGetErrorDatabaseText          | 4-50 |
| XtGetGC                         | 4-50 |
| XtGetKeysymTable                | 4-51 |
| XtGetMultiClickTime             | 4-51 |
| XtGetResourceList               | 4-51 |
| XtGetSelectionRequest           | 4-52 |
| XtGetSelectionTimeout           | 4-52 |
| XtGetSelectionValue             | 4-53 |
| XtGetSelectionValueIncremental  | 4-53 |
| XtGetSelectionValues            | 4-54 |
| XtGetSelectionValuesIncremental | 4-54 |
| XtGetSubresources               | 4-55 |
| XtGetSubvalues                  | 4-55 |
| XtGetValues                     | 4-56 |
| XtGrabButton                    | 4-56 |
| XtGrabKey                       | 4-57 |
| XtGrabKeyboard                  | 4-57 |
| XtGrabPointer                   | 4-58 |
| XtHasCallbacks                  | 4-58 |
| XtInitialize                    | 4-59 |
| XtInitializeWidgetClass         | 4-59 |
| XtInsertEventHandler            | 4-59 |
| XtInsertRawEventHandler         | 4-60 |
| XtInstallAccelerators           | 4-60 |
| XtInstallAllAccelerators        | 4-61 |
| XtIsManaged                     | 4-61 |
| XtIsObject                      | 4-61 |
| XtIsRealized                    | 4-62 |
| XtIsSensitive                   | 4-62 |
| XtIsSubclass                    | 4-62 |
| XtKeysymToKeycodeList           | 4-63 |
| XtLastTimestampProcessed        | 4-63 |
| XtMainLoop                      | 4-63 |
| XtMakeGeometryRequest           | 4-64 |
| XtMakeResizeRequest             | 4-64 |
| XtMalloc                        | 4-64 |
| XtManageChild                   | 4-65 |
| XtManageChildren                | 4-65 |
| XtMenuPopupAction               | 4-65 |



|                           |      |
|---------------------------|------|
| XtMergeArgLists           | 4-66 |
| XtName                    | 4-66 |
| XtNameToWidget            | 4-67 |
| XtNextEvent               | 4-67 |
| XtOpenDisplay             | 4-67 |
| XtOverrideTranslations    | 4-68 |
| XtOwnSelection            | 4-68 |
| XtOwnSelectionIncremental | 4-69 |
| XtParent                  | 4-69 |
| XtParseAcceleratorTable   | 4-70 |
| XtParseTranslationTable   | 4-70 |
| XtPeekEvent               | 4-70 |
| XtPending                 | 4-71 |
| XtPopdown                 | 4-71 |
| XtPopup                   | 4-72 |
| XtPopupSpringLoaded       | 4-72 |
| XtProcessEvent            | 4-72 |
| XtQueryGeometry           | 4-73 |
| XtRealizeWidget           | 4-73 |
| XtRealloc                 | 4-73 |
| XtRegisterCaseConverter   | 4-74 |
| XtRegisterGrabAction      | 4-74 |
| XtReleaseGC               | 4-75 |
| XtRemoveActionHook        | 4-75 |
| XtRemoveAllCallbacks      | 4-75 |
| XtRemoveCallback          | 4-76 |
| XtRemoveCallbacks         | 4-76 |
| XtRemoveEventHandler      | 4-76 |
| XtRemoveGrab              | 4-77 |
| XtRemoveInput             | 4-77 |
| XtRemoveRawEventHandler   | 4-78 |
| XtRemoveTimeOut           | 4-78 |
| XtRemoveWorkProc          | 4-78 |
| XtResolvePathname         | 4-79 |
| XtScreen                  | 4-79 |
| XtScreenDatabase          | 4-80 |
| XtScreenOfObject          | 4-80 |
| XtSetErrorHandler         | 4-80 |
| XtSetErrorMsgHandler      | 4-81 |
| XtSetKeyboardFocus        | 4-81 |
| XtSetKeyTranslator        | 4-81 |
| XtSetLanguageProc         | 4-82 |
| XtSetMappedWhenManaged    | 4-82 |
| XtSetMultiClickTime       | 4-82 |
| XtSetSelectionTimeout     | 4-83 |
| XtSetSensitive            | 4-83 |
| XtSetSubvalues            | 4-84 |

|                                        |      |
|----------------------------------------|------|
| XtSetTypeConverter . . . . .           | 4-84 |
| XtSetValues . . . . .                  | 4-85 |
| XtSetWarningHandler . . . . .          | 4-85 |
| XtSetWarningMsgHandler . . . . .       | 4-85 |
| XtSetWMColormapWindows . . . . .       | 4-86 |
| XtStringConversionWarning . . . . .    | 4-86 |
| XtSuperclass . . . . .                 | 4-86 |
| XtToolkitInitialize . . . . .          | 4-87 |
| XtTranslateCoords . . . . .            | 4-87 |
| XtTranslateKey . . . . .               | 4-87 |
| XtTranslateKeycode . . . . .           | 4-88 |
| XtUngrabButton . . . . .               | 4-88 |
| XtUngrabKey . . . . .                  | 4-89 |
| XtUngrabKeyboard . . . . .             | 4-89 |
| XtUngrabPointer . . . . .              | 4-89 |
| XtUninstallTranslations . . . . .      | 4-90 |
| XtUnmanageChild . . . . .              | 4-90 |
| XtUnmanageChildren . . . . .           | 4-90 |
| XtUnrealizeWidget . . . . .            | 4-91 |
| XtVaAppCreateShell . . . . .           | 4-91 |
| XtVaAppInitialize . . . . .            | 4-91 |
| XtVaCreateArgsList . . . . .           | 4-92 |
| XtVaCreateManagedWidget . . . . .      | 4-92 |
| XtVaCreatePopupShell . . . . .         | 4-93 |
| XtVaCreateWidget . . . . .             | 4-93 |
| XtVaGetApplicationResources . . . . .  | 4-94 |
| XtVaGetSubresources . . . . .          | 4-94 |
| XtVaGetSubvalues . . . . .             | 4-95 |
| XtVaGetValues . . . . .                | 4-95 |
| XtVaSetSubvalues . . . . .             | 4-96 |
| XtVaSetValues . . . . .                | 4-96 |
| XtWarning . . . . .                    | 4-96 |
| XtWarningMsg . . . . .                 | 4-97 |
| XtWidgetToApplicationContext . . . . . | 4-97 |
| XtWindow . . . . .                     | 4-98 |
| XtWindowOfObject . . . . .             | 4-98 |
| XtWindowToWidget . . . . .             | 4-98 |

## 5 Intrinsic Data Structures

|                            |     |
|----------------------------|-----|
| Arg . . . . .              | 5-1 |
| XtActionsRec . . . . .     | 5-1 |
| XtCallbackRec . . . . .    | 5-1 |
| XtConvertArgRec . . . . .  | 5-2 |
| XtI18nContextRec . . . . . | 5-2 |
| XtPopdownIdRec . . . . .   | 5-2 |
| XtResource . . . . .       | 5-2 |

|                         |     |
|-------------------------|-----|
| XtSubstitutionRec ..... | 5-3 |
| XtWidgetGeometry .....  | 5-3 |

## Part III OSF/Motif Toolkit

### 6 OSF/Motif Toolkit Routines

|                                      |      |
|--------------------------------------|------|
| MrmCloseHierarchy .....              | 6-1  |
| MrmFetchBitmapLiteral .....          | 6-1  |
| MrmFetchColorLiteral .....           | 6-2  |
| MrmFetchIconLiteral .....            | 6-2  |
| MrmFetchInterfaceModule .....        | 6-3  |
| MrmFetchLiteral .....                | 6-3  |
| MrmFetchSetValues .....              | 6-4  |
| MrmFetchWidget .....                 | 6-4  |
| MrmFetchWidgetOverride .....         | 6-5  |
| MrmInitialize .....                  | 6-5  |
| MrmOpenHierarchy .....               | 6-5  |
| MrmOpenHierarchyPerDisplay .....     | 6-6  |
| MrmRegisterClass .....               | 6-6  |
| MrmRegisterNames .....               | 6-7  |
| MrmRegisterNamesInHierarchy .....    | 6-7  |
| XmActivateProtocol .....             | 6-7  |
| XmAddProtocolCallback .....          | 6-8  |
| XmAddProtocols .....                 | 6-8  |
| XmAddTabGroup .....                  | 6-9  |
| XmAddToPostFromList .....            | 6-9  |
| XmCascadeButtonGadgetHighlight ..... | 6-9  |
| XmCascadeButtonHighlight .....       | 6-10 |
| XmChangeColor .....                  | 6-10 |
| XmClipboardBeginCopy .....           | 6-10 |
| XmClipboardCancelCopy .....          | 6-11 |
| XmClipboardCopy .....                | 6-11 |
| XmClipboardCopyByName .....          | 6-12 |
| XmClipboardEndCopy .....             | 6-12 |
| XmClipboardEndRetrieve .....         | 6-13 |
| XmClipboardInquireCount .....        | 6-13 |
| XmClipboardInquireFormat .....       | 6-13 |
| XmClipboardInquireLength .....       | 6-14 |
| XmClipboardInquirePendingItems ..... | 6-14 |
| XmClipboardLock .....                | 6-15 |
| XmClipboardRegisterFormat .....      | 6-15 |
| XmClipboardRetrieve .....            | 6-16 |
| XmClipboardStartCopy .....           | 6-16 |
| XmClipboardStartRetrieve .....       | 6-17 |
| XmClipboardUndoCopy .....            | 6-17 |
| XmClipboardUnlock .....              | 6-17 |

|                                       |      |
|---------------------------------------|------|
| XmClipboardWithdrawFormat . . . . .   | 6-18 |
| XmCommandAppendValue . . . . .        | 6-18 |
| XmCommandError . . . . .              | 6-19 |
| XmCommandGetChild . . . . .           | 6-19 |
| XmCommandSetValue . . . . .           | 6-19 |
| XmConvertUnits . . . . .              | 6-20 |
| XmCreateArrowButton . . . . .         | 6-20 |
| XmCreateArrowButtonGadget . . . . .   | 6-21 |
| XmCreateBulletinBoard . . . . .       | 6-21 |
| XmCreateBulletinBoardDialog . . . . . | 6-21 |
| XmCreateCascadeButton . . . . .       | 6-22 |
| XmCreateCascadeButtonGadget . . . . . | 6-22 |
| XmCreateCommand . . . . .             | 6-23 |
| XmCreateCommandDialog . . . . .       | 6-23 |
| XmCreateDialogShell . . . . .         | 6-24 |
| XmCreateDragIcon . . . . .            | 6-24 |
| XmCreateDrawingArea . . . . .         | 6-24 |
| XmCreateDrawnButton . . . . .         | 6-25 |
| XmCreateErrorDialog . . . . .         | 6-25 |
| XmCreateFileSelectionBox . . . . .    | 6-26 |
| XmCreateFileSelectionDialog . . . . . | 6-26 |
| XmCreateForm . . . . .                | 6-27 |
| XmCreateFormDialog . . . . .          | 6-27 |
| XmCreateFrame . . . . .               | 6-27 |
| XmCreateInformationDialog . . . . .   | 6-28 |
| XmCreateLabel . . . . .               | 6-28 |
| XmCreateLabelGadget . . . . .         | 6-29 |
| XmCreateList . . . . .                | 6-29 |
| XmCreateMainWindow . . . . .          | 6-30 |
| XmCreateMenuBar . . . . .             | 6-30 |
| XmCreateMenuShell . . . . .           | 6-30 |
| XmCreateMessageBox . . . . .          | 6-31 |
| XmCreateMessageDialog . . . . .       | 6-31 |
| XmCreateOptionsMenu . . . . .         | 6-32 |
| XmCreatePanedWindow . . . . .         | 6-32 |
| XmCreatePopupMenu . . . . .           | 6-33 |
| XmCreatePromptDialog . . . . .        | 6-33 |
| XmCreatePulldownMenu . . . . .        | 6-33 |
| XmCreatePushButton . . . . .          | 6-34 |
| XmCreatePushButtonGadget . . . . .    | 6-34 |
| XmCreateQuestionDialog . . . . .      | 6-35 |
| XmCreateRadioBox . . . . .            | 6-35 |
| XmCreateRowColumn . . . . .           | 6-36 |
| XmCreateScale . . . . .               | 6-36 |
| XmCreateScrollBar . . . . .           | 6-36 |
| XmCreateScrolledList . . . . .        | 6-37 |
| XmCreateScrolledText . . . . .        | 6-37 |

|                                  |      |
|----------------------------------|------|
| XmCreateScrolledWindow           | 6-38 |
| XmCreateSelectionBox             | 6-38 |
| XmCreateSelectionDialog          | 6-39 |
| XmCreateSeparator                | 6-39 |
| XmCreateSeparatorGadget          | 6-39 |
| XmCreateSimpleCheckBox           | 6-40 |
| XmCreateSimpleMenuBar            | 6-40 |
| XmCreateSimpleOptionMenu         | 6-41 |
| XmCreateSimplePopupMenu          | 6-41 |
| XmCreateSimplePulldownMenu       | 6-42 |
| XmCreatePushButtonGadget         | 6-42 |
| XmCreateSimpleRadioBox           | 6-42 |
| XmCreateTemplateDialog           | 6-43 |
| XmCreateText                     | 6-43 |
| XmCreateTextField                | 6-44 |
| XmCreateToggleButton             | 6-44 |
| XmCreateToggleButtonGadget       | 6-45 |
| XmCreateWarningDialog            | 6-45 |
| XmCreateWorkArea                 | 6-45 |
| XmCreateWorkingDialog            | 6-46 |
| XmCreateWorkArea                 | 6-46 |
| XmCreateWorkingDialog            | 6-47 |
| XmCvtCTToXmString                | 6-47 |
| XmCvtStringToUnitType            | 6-48 |
| XmCvtXmStringToCT                | 6-48 |
| XmDeactivateProtocol             | 6-48 |
| XmDestroyPixmap                  | 6-49 |
| XmDragCancel                     | 6-49 |
| XmDragStart                      | 6-49 |
| XmDropSiteConfigureStackingOrder | 6-50 |
| XmDropSiteEndUpdate              | 6-50 |
| XmDropSiteGetActiveVisuals       | 6-51 |
| XmDropSiteQueryStackingOrder     | 6-51 |
| XmDropSiteRegister               | 6-51 |
| XmDropSiteRetrieve               | 6-52 |
| XmDropSiteStartUpdate            | 6-52 |
| XmDropSiteUnregister             | 6-53 |
| XmDropSiteUpdate                 | 6-53 |
| XmDropTransferAdd                | 6-53 |
| XmDropTransferStart              | 6-54 |
| XmFileSelectionBoxGetChild       | 6-54 |
| XmFileSelectionDoSearch          | 6-54 |
| XmFontListAdd                    | 6-55 |
| XmFontListAppendEntry            | 6-55 |
| XmFontListCopy                   | 6-56 |
| XmFontListCreate                 | 6-56 |
| XmFontListEntryCreate            | 6-56 |

|                           |      |
|---------------------------|------|
| XmFontListEntryFree       | 6-57 |
| XmFontListEntryGetFont    | 6-57 |
| XmFontListEntryGetTag     | 6-57 |
| XmFontListEntryLoad       | 6-58 |
| XmFontListFree            | 6-58 |
| XmFontListFreeFontContext | 6-59 |
| XmFontListGetNextFont     | 6-59 |
| XmFontListInitFontContext | 6-59 |
| XmFontListNextEntry       | 6-60 |
| XmFontListRemoveEntry     | 6-60 |
| XmGetAtomName             | 6-60 |
| XmGetColorCalculation     | 6-61 |
| XmGetColors               | 6-61 |
| XmGetDestination          | 6-62 |
| XmGetDragContext          | 6-62 |
| XmGetFocusWidget          | 6-62 |
| XmGetMenuCursor           | 6-63 |
| XmGetPixmap               | 6-63 |
| XmGetPixmapByDepth        | 6-63 |
| XmGetPostedFromWidget     | 6-64 |
| XmGetTabGroup             | 6-64 |
| XmGetTearOffControl       | 6-65 |
| XmGetVisibility           | 6-65 |
| XmGetXmDisplay            | 6-65 |
| XmGetXmScreen             | 6-66 |
| XmImGetXIM                | 6-66 |
| XmImMbLookupString        | 6-66 |
| XmImRegister              | 6-67 |
| XmImSetFocusValues        | 6-67 |
| XmImSetValues             | 6-68 |
| XmImUnregister            | 6-68 |
| XmImUnsetFocus            | 6-68 |
| XmImVaSetFocusValues      | 6-69 |
| XmImVaSetValues           | 6-69 |
| XmInstallImage            | 6-69 |
| XmInternAtom              | 6-70 |
| XmIsMotifWMRunning        | 6-70 |
| XmIsTraversable           | 6-70 |
| XmListAddItem             | 6-71 |
| XmListAddItems            | 6-71 |
| XmListAddItemsUnselected  | 6-72 |
| XmListAddItemUnselected   | 6-72 |
| XmListDeleteAllItems      | 6-72 |
| XmListDeleteItem          | 6-73 |
| XmListDeleteItems         | 6-73 |
| XmListDeleteItemsPos      | 6-74 |
| XmListDeletePos           | 6-74 |

|                                           |      |
|-------------------------------------------|------|
| XmListDeletePositions . . . . .           | 6-74 |
| XmListDeselectAllItems . . . . .          | 6-75 |
| XmListDeselectItem . . . . .              | 6-75 |
| XmListDeselectPos . . . . .               | 6-75 |
| XmListGetKbdItemPos . . . . .             | 6-76 |
| XmListGetMatchPos . . . . .               | 6-76 |
| XmListGetSelectedPos . . . . .            | 6-77 |
| XmListItemExists . . . . .                | 6-77 |
| XmListItemPos . . . . .                   | 6-77 |
| XmListPosSelected . . . . .               | 6-78 |
| XmListPosToBounds . . . . .               | 6-78 |
| XmListReplaceItems . . . . .              | 6-79 |
| XmListReplaceItemsPos . . . . .           | 6-79 |
| XmListReplaceItemsPosUnselected . . . . . | 6-79 |
| XmListReplaceItemsUnselected . . . . .    | 6-80 |
| XmListReplacePositions . . . . .          | 6-80 |
| XmListSelectItem . . . . .                | 6-81 |
| XmListSelectPos . . . . .                 | 6-81 |
| XmListSetAddMode . . . . .                | 6-81 |
| XmListSetBottomItem . . . . .             | 6-82 |
| XmListSetBottomPos . . . . .              | 6-82 |
| XmListSetHorizPos . . . . .               | 6-82 |
| XmListSetItem . . . . .                   | 6-83 |
| XmListSetKbdItemPos . . . . .             | 6-83 |
| XmListSetPos . . . . .                    | 6-84 |
| XmListUpdateSelectedList . . . . .        | 6-84 |
| XmListYToPos . . . . .                    | 6-84 |
| XmMainWindowSep1 . . . . .                | 6-85 |
| XmMainWindowSep2 . . . . .                | 6-85 |
| XmMainWindowSep3 . . . . .                | 6-85 |
| XmMainWindowSetAreas . . . . .            | 6-86 |
| XmMapSegmentEncoding . . . . .            | 6-86 |
| XmMenuPosition . . . . .                  | 6-86 |
| XmMessageBoxGetChild . . . . .            | 6-87 |
| XmOptionButtonGadget . . . . .            | 6-87 |
| XmOptionLabelGadget . . . . .             | 6-88 |
| XmProcessTraversal . . . . .              | 6-88 |
| XmRegisterConverters . . . . .            | 6-88 |
| XmRegisterSegmentEncoding . . . . .       | 6-88 |
| XmRemoveFromPostFromList . . . . .        | 6-89 |
| XmRemoveProtocolCallback . . . . .        | 6-89 |
| XmRemoveProtocols . . . . .               | 6-90 |
| XmRemoveTabGroup . . . . .                | 6-90 |
| XmRepTypeAddReverse . . . . .             | 6-90 |
| XmRepTypeGetId . . . . .                  | 6-91 |
| XmRepTypeGetNameList . . . . .            | 6-91 |
| XmRepTypeGetRecord . . . . .              | 6-91 |

|                                       |       |
|---------------------------------------|-------|
| XmRepTypeGetRegistered                | 6-92  |
| XmRepTypeInstallTearOffModelConverter | 6-92  |
| XmRepTypeRegister                     | 6-92  |
| XmRepTypeValidValue                   | 6-92  |
| XmResolveAllPartOffsets               | 6-93  |
| XmResolvePartOffsets                  | 6-93  |
| XmScaleGetValue                       | 6-94  |
| XmScaleSetValue                       | 6-94  |
| XmScrollBarGetValues                  | 6-94  |
| XmScrollBarSetValues                  | 6-95  |
| XmScrollVisible                       | 6-95  |
| XmScrolledWindowSetAreas              | 6-96  |
| XmSelectionBoxGetChild                | 6-96  |
| XmSetColorCalculation                 | 6-96  |
| XmSetFontUnit                         | 6-97  |
| XmSetFontUnits                        | 6-97  |
| XmSetMenuCursor                       | 6-98  |
| XmSetProtocolHooks                    | 6-98  |
| XmStringBaseline                      | 6-98  |
| XmStringByteCompare                   | 6-99  |
| XmStringCompare                       | 6-99  |
| XmStringConcat                        | 6-100 |
| XmStringCopy                          | 6-100 |
| XmStringCreate                        | 6-100 |
| XmStringCreateLocalized               | 6-101 |
| XmStringCreateLtoR                    | 6-101 |
| XmStringCreateSimple                  | 6-101 |
| XmStringDirectionCreate               | 6-102 |
| XmStringDraw                          | 6-102 |
| XmStringDrawImage                     | 6-103 |
| XmStringDrawUnderline                 | 6-103 |
| XmStringEmpty                         | 6-104 |
| XmStringExtent                        | 6-104 |
| XmStringFree                          | 6-105 |
| XmStringFreeContext                   | 6-105 |
| XmStringGetLtoR                       | 6-105 |
| XmStringGetNextComponent              | 6-106 |
| XmStringGetNextSegment                | 6-106 |
| XmStringHasSubstring                  | 6-107 |
| XmStringHeight                        | 6-107 |
| XmStringInitContext                   | 6-107 |
| XmStringLength                        | 6-108 |
| XmStringLineCount                     | 6-108 |
| XmStringNConcat                       | 6-108 |
| XmStringNCopy                         | 6-109 |
| XmStringPeekNextComponent             | 6-109 |
| XmStringSegmentCreate                 | 6-110 |



|                                 |       |
|---------------------------------|-------|
| XmStringSeparatorCreate         | 6-110 |
| XmStringWidth                   | 6-110 |
| XmTargetsAreCompatible          | 6-111 |
| XmTextClearSelection            | 6-111 |
| XmTextCopy                      | 6-112 |
| XmTextCut                       | 6-112 |
| XmTextDisableRedisplay          | 6-112 |
| XmTextEnableRedisplay           | 6-113 |
| XmTextFieldClearSelection       | 6-113 |
| XmTextFieldCopy                 | 6-113 |
| XmTextFieldCut                  | 6-114 |
| XmTextFieldGetBaseline          | 6-114 |
| XmTextFieldGetCursorPosition    | 6-114 |
| XmTextFieldGetEditable          | 6-115 |
| XmTextFieldGetInsertionPosition | 6-115 |
| XmTextFieldGetLastPosition      | 6-115 |
| XmTextFieldGetMaxLength         | 6-116 |
| XmTextFieldGetSelection         | 6-116 |
| XmTextFieldGetSelectionPosition | 6-116 |
| XmTextFieldGetSelectionWcs      | 6-117 |
| XmTextFieldGetString            | 6-117 |
| XmTextFieldGetStringWcs         | 6-117 |
| XmTextFieldGetSubstring         | 6-118 |
| XmTextFieldGetSubstringWcs      | 6-118 |
| XmTextFieldInsert               | 6-119 |
| XmTextFieldInsertWcs            | 6-119 |
| XmTextFieldPaste                | 6-119 |
| XmTextFieldPosToXY              | 6-120 |
| XmTextFieldRemove               | 6-120 |
| XmTextFieldReplace              | 6-121 |
| XmTextFieldReplaceWcs           | 6-121 |
| XmTextFieldSetAddMode           | 6-121 |
| XmTextFieldSetCursorPosition    | 6-122 |
| XmTextFieldSetEditable          | 6-122 |
| XmTextFieldSetHighlight         | 6-123 |
| XmTextFieldSetInsertionPosition | 6-123 |
| XmTextFieldSetMaxLength         | 6-123 |
| XmTextFieldSetSelection         | 6-124 |
| XmTextFieldSetString            | 6-124 |
| XmTextFieldShowPosition         | 6-124 |
| XmTextFieldXYToPos              | 6-125 |
| XmTextFindString                | 6-125 |
| XmTextFindStringWcs             | 6-126 |
| XmTextGetBaseline               | 6-126 |
| XmTextGetCursorPosition         | 6-126 |
| XmTextGetEditable               | 6-127 |
| XmTextGetInsertionPosition      | 6-127 |

|                                        |       |
|----------------------------------------|-------|
| XmTextGetLastPosition . . . . .        | 6-128 |
| XmTextGetMaxLength . . . . .           | 6-128 |
| XmTextGetSelection . . . . .           | 6-128 |
| XmTextGetSelectionPosition . . . . .   | 6-129 |
| XmTextGetSelectionWcs . . . . .        | 6-129 |
| XmTextGetSource . . . . .              | 6-129 |
| XmTextGetString . . . . .              | 6-130 |
| XmTextGetStringWcs . . . . .           | 6-130 |
| XmTextGetSubstring . . . . .           | 6-130 |
| XmTextGetSubstringWcs . . . . .        | 6-131 |
| XmTextGetTopCharacter . . . . .        | 6-131 |
| XmTextInsert . . . . .                 | 6-132 |
| XmTextInsertWcs . . . . .              | 6-132 |
| XmTextPaste . . . . .                  | 6-132 |
| XmTextPosToXY . . . . .                | 6-133 |
| XmTextRemove . . . . .                 | 6-133 |
| XmTextReplace . . . . .                | 6-134 |
| XmTextReplaceWcs . . . . .             | 6-134 |
| XmTextScroll . . . . .                 | 6-134 |
| XmTextSetAddMode . . . . .             | 6-135 |
| XmTextSetCursorPosition . . . . .      | 6-135 |
| XmTextSetEditable . . . . .            | 6-136 |
| XmTextSetHighlight . . . . .           | 6-136 |
| XmTextSetInsertionPosition . . . . .   | 6-136 |
| XmTextSetMaxLength . . . . .           | 6-137 |
| XmTextSetSelection . . . . .           | 6-137 |
| XmTextSetSource . . . . .              | 6-137 |
| XmTextSetString . . . . .              | 6-138 |
| XmTextSetStringWcs . . . . .           | 6-138 |
| XmTextSetTopCharacter . . . . .        | 6-139 |
| XmTextShowPosition . . . . .           | 6-139 |
| XmTextXYToPos . . . . .                | 6-139 |
| XmToggleButtonGadgetGetState . . . . . | 6-140 |
| XmToggleButtonGadgetSetState . . . . . | 6-140 |
| XmToggleButtonGetState . . . . .       | 6-140 |
| XmToggleButtonSetState . . . . .       | 6-141 |
| XmTrackingEvent . . . . .              | 6-141 |
| XmTrackingLocate . . . . .             | 6-142 |
| XmTranslateKey . . . . .               | 6-142 |
| XmUninstallImage . . . . .             | 6-142 |
| XmUpdateDisplay . . . . .              | 6-143 |
| XmVaCreateSimpleCheckBox . . . . .     | 6-143 |
| XmVaCreateSimpleMenuBar . . . . .      | 6-144 |
| XmVaCreateSimpleOptionsMenu . . . . .  | 6-144 |
| XmVaCreateSimplePopupMenu . . . . .    | 6-144 |
| XmVaCreateSimplePulldownMenu . . . . . | 6-145 |
| XmVaCreateSimpleRadioBox . . . . .     | 6-145 |

|                              |       |
|------------------------------|-------|
| XmWidgetGetBaselines .....   | 6-146 |
| XmWidgetGetDisplayRect ..... | 6-146 |

## 7 OSF/Motif Toolkit Data Structures

|                                        |      |
|----------------------------------------|------|
| MotifWmHints .....                     | 7-1  |
| MotifWmInfo .....                      | 7-1  |
| MrmOsOpenParam .....                   | 7-1  |
| MrmRegisterArg .....                   | 7-2  |
| PropMotifWmHints .....                 | 7-2  |
| PropMotifWmInfo .....                  | 7-2  |
| XmAnyCallbackStruct .....              | 7-3  |
| XmAnyICCCallbackStruct .....           | 7-3  |
| XmArrowButtonCallbackStruct .....      | 7-3  |
| XmClipboardPendingRec .....            | 7-3  |
| XmCommandCallbackStruct .....          | 7-4  |
| XmDragDropFinishCallbackStruct .....   | 7-4  |
| XmDropProcCallbackStruct .....         | 7-4  |
| XmDragMotionCallbackStruct .....       | 7-5  |
| XmDragProcCallbackStruct .....         | 7-5  |
| XmDrawingAreaCallbackStruct .....      | 7-5  |
| XmDrawnButtonCallbackStruct .....      | 7-6  |
| XmDropFinishCallbackStruct .....       | 7-6  |
| XmDropSiteEnterCallbackStruct .....    | 7-6  |
| XmDropSiteLeaveCallbackStruct .....    | 7-7  |
| XmDropSiteVisualsRec .....             | 7-7  |
| XmDropStartCallbackStruct .....        | 7-8  |
| XmDropTransferEntryRec .....           | 7-8  |
| XmFileSelectionCallbackStruct .....    | 7-8  |
| XmListCallbackStruct .....             | 7-9  |
| XmOperationChangedCallbackStruct ..... | 7-9  |
| XmPushButtonCallbackStruct .....       | 7-10 |
| XmRepTypeEntryRec .....                | 7-10 |
| XmRepTypeListRec .....                 | 7-10 |
| XmRowColumnCallbackStruct .....        | 7-11 |
| XmScaleCallbackStruct .....            | 7-11 |
| XmScrollBarCallbackStruct .....        | 7-11 |
| XmSecondaryResourceDataRec .....       | 7-12 |
| XmSelectionBoxCallbackStruct .....     | 7-12 |
| XmTextBlockRec .....                   | 7-12 |
| XmTextBlockRecWcs .....                | 7-13 |
| XmTextVerifyCallbackStruct .....       | 7-13 |
| XmTextVerifyCallbackStructWcs .....    | 7-13 |
| XmToggleButtonCallbackStruct .....     | 7-14 |
| XmTopLevelEnterCallbackStruct .....    | 7-14 |
| XmTopLevelLeaveCallbackStruct .....    | 7-14 |
| XmTraverseObscuredCallbackStruct ..... | 7-15 |

## Part IV DECwindows Toolkit Extensions

### 8 DECwindows Toolkit Extensions Routines

|                                         |      |
|-----------------------------------------|------|
| DXmActivateWidget . . . . .             | 8-1  |
| DXmChangeWindowGeometry . . . . .       | 8-1  |
| DXmMakeGeometryRequest . . . . .        | 8-2  |
| DXmChildren . . . . .                   | 8-2  |
| DXmColorMixGetNewColor . . . . .        | 8-2  |
| DXmColorMixSetNewColor . . . . .        | 8-3  |
| DXmCreateColorMix . . . . .             | 8-3  |
| DXmCreateColorMixDialog . . . . .       | 8-4  |
| DXmCreateCSText . . . . .               | 8-4  |
| DXmCreateCursor . . . . .               | 8-4  |
| DXmCreateHelp . . . . .                 | 8-5  |
| DXmCreateHelpDialog . . . . .           | 8-5  |
| DXmCreatePrintBox . . . . .             | 8-6  |
| DXmCreatePrintDialog . . . . .          | 8-6  |
| DXmCreateScrolledCSText . . . . .       | 8-6  |
| DXmCreateSvn . . . . .                  | 8-7  |
| DXmCSContainsStringCharSet . . . . .    | 8-7  |
| DXmCSTextClearSelection . . . . .       | 8-8  |
| DXmCSTextCopy . . . . .                 | 8-8  |
| DXmCSTextCut . . . . .                  | 8-8  |
| DXmCSTextDisableRedisplay . . . . .     | 8-9  |
| DXmCSTextEnableRedisplay . . . . .      | 8-9  |
| DXmCSTextGetEditable . . . . .          | 8-9  |
| DXmCSTextGetInsertionPosition . . . . . | 8-10 |
| DXmCSTextGetLastPosition . . . . .      | 8-10 |
| DXmCSTextGetMaxLength . . . . .         | 8-11 |
| DXmCSTextGetSelection . . . . .         | 8-11 |
| DXmCSTextGetSelectionInfo . . . . .     | 8-11 |
| DXmCSTextGetString . . . . .            | 8-12 |
| DXmCSTextGetStringWrapped . . . . .     | 8-12 |
| DXmCSTextGetTextPath . . . . .          | 8-12 |
| DXmCSTextGetTopPosition . . . . .       | 8-13 |
| DXmCSTextHasSelection . . . . .         | 8-13 |
| DXmCSTextHorizontalScroll . . . . .     | 8-14 |
| DXmCSTextInsert . . . . .               | 8-14 |
| DXmCSTextMarkRedraw . . . . .           | 8-14 |
| DXmCSTextGetInsertionPosition . . . . . | 8-15 |
| DXmCSTextInvalidate . . . . .           | 8-15 |
| DXmCSTextNumLines . . . . .             | 8-15 |
| DXmCSTextPaste . . . . .                | 8-16 |
| DXmCSTextPosToLine . . . . .            | 8-16 |
| DXmCSTextPosToXY . . . . .              | 8-17 |
| DXmCSTextRemove . . . . .               | 8-17 |

|                                         |      |
|-----------------------------------------|------|
| DXmCSTextRead . . . . .                 | 8-17 |
| DXmCSTextReplace . . . . .              | 8-18 |
| DXmCSTextSetAddMode . . . . .           | 8-18 |
| DXmCSTextSetEditable . . . . .          | 8-19 |
| DXmCSTextSetHighlight . . . . .         | 8-19 |
| DXmCSTextSetInsertionPosition . . . . . | 8-19 |
| DXmCSTextSetMaxLength . . . . .         | 8-20 |
| DXmCSTextSetSelection . . . . .         | 8-20 |
| DXmCSTextSetString . . . . .            | 8-20 |
| DXmCSTextSetTopPosition . . . . .       | 8-21 |
| DXmCSTextShowPosition . . . . .         | 8-21 |
| DXmCSTextVerticalScroll . . . . .       | 8-22 |
| DXmCSTextXYToPos . . . . .              | 8-22 |
| DXmCvtCStoDDIF . . . . .                | 8-22 |
| DXmCvtCStoFC . . . . .                  | 8-23 |
| DXmCvtCStoOS . . . . .                  | 8-23 |
| DXmCvtDDIFtoCS . . . . .                | 8-24 |
| DXmCvtFCtoCS . . . . .                  | 8-24 |
| DXmCvtOStoCS . . . . .                  | 8-24 |
| DXmDisplayCSMessage . . . . .           | 8-25 |
| DXmDisplayVmsMessage . . . . .          | 8-25 |
| DXmFindFontFallback . . . . .           | 8-26 |
| DXmFontListCreateDefault . . . . .      | 8-26 |
| DXmFormSpaceButtonsEqually . . . . .    | 8-27 |
| DXmGetLocaleCharset . . . . .           | 8-27 |
| DXmGetLocaleCharsets . . . . .          | 8-28 |
| DXmGetLocaleMnemonic . . . . .          | 8-28 |
| DXmGetLocaleString . . . . .            | 8-28 |
| DXmHelpOnContext . . . . .              | 8-29 |
| DXmHelpSystemClose . . . . .            | 8-29 |
| DXmHelpSystemDisplay . . . . .          | 8-29 |
| DXmHelpSystemOpen . . . . .             | 8-30 |
| DXmInitialize . . . . .                 | 8-30 |
| DXmLoadQueryFont . . . . .              | 8-30 |
| DXmNumChildren . . . . .                | 8-31 |
| DXmPositionWidget . . . . .             | 8-31 |
| DXmPrintWgtAugmentList . . . . .        | 8-32 |
| DXmPrintWgtPrintJob . . . . .           | 8-32 |
| DXmStringCheck . . . . .                | 8-32 |
| DXmSvnAddEntries . . . . .              | 8-33 |
| DXmSvnAutoScrollCheck . . . . .         | 8-33 |
| DXmSvnAutoScrollDisplay . . . . .       | 8-34 |
| DXmSvnClearHighlight . . . . .          | 8-34 |
| DXmSvnClearHighlighting . . . . .       | 8-34 |
| DXmSvnClearSelection . . . . .          | 8-35 |
| DXmSvnClearSelections . . . . .         | 8-35 |
| DXmSvnDeleteEntries . . . . .           | 8-35 |

|                                        |      |
|----------------------------------------|------|
| DXmSvnDisableDisplay . . . . .         | 8-36 |
| DXmSvnEnableDisplay . . . . .          | 8-36 |
| DXmSvnFlushEntry . . . . .             | 8-36 |
| DXmSvnGetComponentNumber . . . . .     | 8-37 |
| DXmSvnGetComponentTag . . . . .        | 8-37 |
| DXmSvnGetComponentWidth . . . . .      | 8-37 |
| DXmSvnGetComponentText . . . . .       | 8-38 |
| DXmSvnGetDisplayed . . . . .           | 8-38 |
| DXmSvnGetEntryLevel . . . . .          | 8-39 |
| DXmSvnGetEntryNumber . . . . .         | 8-39 |
| DXmSvnGetEntryPosition . . . . .       | 8-39 |
| DXmSvnGetEntrySensitivity . . . . .    | 8-40 |
| DXmSvnGetEntryTag . . . . .            | 8-40 |
| DXmSvnGetHighlighted . . . . .         | 8-41 |
| DXmSvnGetNumDisplayed . . . . .        | 8-41 |
| DXmSvnGetNumHighlighted . . . . .      | 8-41 |
| DXmSvnGetNumSelections . . . . .       | 8-42 |
| DXmSvnGetPrimaryWorkWidget . . . . .   | 8-42 |
| DXmSvnGetSecondaryWorkWidget . . . . . | 8-43 |
| DXmSvnGetSelections . . . . .          | 8-43 |
| DXmSvnGetTreePosition . . . . .        | 8-43 |
| DXmSvnHideHighlighting . . . . .       | 8-44 |
| DXmSvnHideSelections . . . . .         | 8-44 |
| DXmSvnHighlightAll . . . . .           | 8-44 |
| DXmSvnHighlightEntry . . . . .         | 8-45 |
| DXmSvnInsertComponent . . . . .        | 8-45 |
| DXmSvnInvalidateEntry . . . . .        | 8-46 |
| DXmSvnMapPosition . . . . .            | 8-46 |
| DXmSvnPosition Display . . . . .       | 8-46 |
| DXmSvnRemoveComponent . . . . .        | 8-47 |
| DXmSvnSelectAll . . . . .              | 8-47 |
| DXmSvnSelectComponent . . . . .        | 8-48 |
| DXmSvnSelectEntry . . . . .            | 8-48 |
| DXmSvnSetApplDragging . . . . .        | 8-48 |
| DXmSvnSetComponentHidden . . . . .     | 8-49 |
| DXmSvnSetComponentPixmap . . . . .     | 8-49 |
| DXmSvnSetComponentTag . . . . .        | 8-50 |
| DXmSvnSetComponentText . . . . .       | 8-50 |
| DXmSvnSetComponentWidget . . . . .     | 8-51 |
| DXmSvnSetComponentWidth . . . . .      | 8-51 |
| DXmSvnSetEntry . . . . .               | 8-51 |
| DXmSvnSetEntryIndexWindow . . . . .    | 8-52 |
| DXmSvnSetEntryNumComponents . . . . .  | 8-52 |
| DXmSvnSetEntryPosition . . . . .       | 8-53 |
| DXmSvnSetEntrySensitivity . . . . .    | 8-53 |
| DXmSvnSetEntryTag . . . . .            | 8-54 |
| DXmSvnShowHighlighting . . . . .       | 8-54 |

|                             |      |
|-----------------------------|------|
| DXmSvnSetTreePosition ..... | 8-54 |
| DXmSvnShowSelections .....  | 8-55 |
| DXmSvnValidateAll .....     | 8-55 |
| DXmSvnWidget .....          | 8-55 |

## 9 DECwindows Toolkit Extensions Data Structures

|                                     |     |
|-------------------------------------|-----|
| DXmColorMixCallbackStruct .....     | 9-1 |
| DXmCSTextCallbackStruct .....       | 9-1 |
| DXmCSTextVerifyCallbackStruct ..... | 9-2 |
| DXmPrintFormatStruct .....          | 9-2 |
| DxmPrintOptionsMenuStruct .....     | 9-2 |
| DXmSvnCallbackStruct .....          | 9-3 |

## Index

### Figures

|      |                                              |      |
|------|----------------------------------------------|------|
| 3-1  | Any Event Data Structure .....               | 3-1  |
| 3-2  | Arc Data Structure .....                     | 3-2  |
| 3-3  | Button Event Data Structure .....            | 3-3  |
| 3-4  | Char 2B Data Structure .....                 | 3-4  |
| 3-5  | Char Struct Data Structure .....             | 3-5  |
| 3-6  | Circulate Event Data Structure .....         | 3-5  |
| 3-7  | Circulate Request Event Data Structure ..... | 3-6  |
| 3-8  | Class Hint Data Structure .....              | 3-7  |
| 3-9  | Client Message Event Data Structure .....    | 3-7  |
| 3-10 | Color Data Structure .....                   | 3-8  |
| 3-11 | Color Map Event Data Structure .....         | 3-9  |
| 3-12 | Compose Status Data Structure .....          | 3-10 |
| 3-13 | Configure Event Data Structure .....         | 3-11 |
| 3-14 | Configure Event Data Structure .....         | 3-12 |
| 3-15 | Create Window Event Data Structure .....     | 3-14 |
| 3-16 | Crossing Event Data Structure .....          | 3-15 |
| 3-17 | Destroy Window Event Data Structure .....    | 3-17 |
| 3-18 | Error Event Data Structure .....             | 3-18 |
| 3-19 | Event Data Structure .....                   | 3-18 |
| 3-20 | Expose Event Data Structure .....            | 3-20 |
| 3-21 | Ext Codes Data Structure .....               | 3-21 |
| 3-22 | Ext Codes Data Structure .....               | 3-21 |
| 3-23 | Focus Change Event Data Structure .....      | 3-22 |
| 3-24 | Font Property Data Structure .....           | 3-23 |
| 3-25 | Font Data Structure .....                    | 3-24 |
| 3-26 | GC Values Data Structure .....               | 3-26 |
| 3-27 | Graphics Expose Event Data Structure .....   | 3-29 |
| 3-28 | Gravity Event Data Structure .....           | 3-30 |
| 3-29 | Host Address Data Structure .....            | 3-31 |

|      |                                                  |      |
|------|--------------------------------------------------|------|
| 3-30 | Icon Size Data Structure . . . . .               | 3-32 |
| 3-31 | Image Data Structure . . . . .                   | 3-33 |
| 3-32 | Key Event Data Structure . . . . .               | 3-35 |
| 3-33 | Keyboard Control Data Structure . . . . .        | 3-36 |
| 3-34 | Keyboard State Data Structure . . . . .          | 3-37 |
| 3-35 | Keymap Event Data Structure . . . . .            | 3-38 |
| 3-36 | Map Event Data Structure . . . . .               | 3-39 |
| 3-37 | Map Request Event Data Structure . . . . .       | 3-40 |
| 3-38 | Mapping Event Data Structure . . . . .           | 3-40 |
| 3-39 | Modifier Keymap Data Structure . . . . .         | 3-41 |
| 3-40 | Motion Event Data Structure . . . . .            | 3-42 |
| 3-41 | No Expose Event Data Structure . . . . .         | 3-43 |
| 3-42 | Pixmap Format Values Data Structure . . . . .    | 3-44 |
| 3-43 | Point Data Structure . . . . .                   | 3-45 |
| 3-44 | Property Event Data Structure . . . . .          | 3-45 |
| 3-45 | Rectangle Data Structure . . . . .               | 3-46 |
| 3-46 | Reparent Event Data Structure . . . . .          | 3-47 |
| 3-47 | Resize Request Event Data Structure . . . . .    | 3-48 |
| 3-48 | Resource Manager Option Data Structure . . . . . | 3-49 |
| 3-49 | Resource Manager Value Data Structure . . . . .  | 3-49 |
| 3-50 | Segment Data Structure . . . . .                 | 3-50 |
| 3-51 | Selection Clear Event Data Structure . . . . .   | 3-51 |
| 3-52 | Selection Event Data Structure . . . . .         | 3-51 |
| 3-53 | Selection Request Event Data Structure . . . . . | 3-52 |
| 3-54 | Set Window Attributes Data Structure . . . . .   | 3-53 |
| 3-55 | Size Hints Data Structure . . . . .              | 3-55 |
| 3-56 | Standard Color Map Data Structure . . . . .      | 3-57 |
| 3-57 | Text Item Data Structure . . . . .               | 3-58 |
| 3-58 | Text Item 16 Data Structure . . . . .            | 3-58 |
| 3-59 | Text Property Data Structure . . . . .           | 3-59 |
| 3-60 | Time Coordinate Data Structure . . . . .         | 3-60 |
| 3-61 | Unmap Event Data Structure . . . . .             | 3-60 |
| 3-62 | Visibility Event Data Structure . . . . .        | 3-61 |
| 3-63 | Visual Info Data Structure . . . . .             | 3-62 |
| 3-64 | Window Attributes Data Structure . . . . .       | 3-63 |
| 3-65 | Window Changes Data Structure . . . . .          | 3-66 |
| 3-66 | WM Hints Data Structure . . . . .                | 3-67 |

## Tables

|     |                                   |      |
|-----|-----------------------------------|------|
| 1-1 | General Rules of Syntax . . . . . | 1-1  |
| 1-2 | OpenVMS Usage Entries . . . . .   | 1-2  |
| 1-3 | Access Entries . . . . .          | 1-4  |
| 1-4 | Mechanism Entries . . . . .       | 1-5  |
| 1-5 | Syntax Conventions . . . . .      | 1-5  |
| 1-6 | Type Entries . . . . .            | 1-6  |
| 1-7 | Access Entries . . . . .          | 1-10 |



|      |                                                          |       |
|------|----------------------------------------------------------|-------|
| 1-8  | Mechanism Entries . . . . .                              | 1-11  |
| 2-1  | Event Mask Description . . . . .                         | 2-12  |
| 2-2  | Change Mask Bits . . . . .                               | 2-21  |
| 2-3  | Parse Mask Bits . . . . .                                | 2-67  |
| 2-4  | Event Mask Description . . . . .                         | 2-101 |
| 2-5  | Event Mask Description . . . . .                         | 2-105 |
| 2-6  | Parse Mask Bits . . . . .                                | 2-132 |
| 2-7  | Graphics Context Codes for Function Member . . . . .     | 2-183 |
| 2-8  | Graphics Context Codes for Function Member . . . . .     | 2-193 |
| 3-1  | Any Event Data Structure Members . . . . .               | 3-2   |
| 3-2  | Arc Data Structure Members . . . . .                     | 3-2   |
| 3-3  | Button Event Data Structure Members . . . . .            | 3-3   |
| 3-4  | Char 2B Data Structure Members . . . . .                 | 3-4   |
| 3-5  | Char Struct Data Structure Members . . . . .             | 3-5   |
| 3-6  | Circulate Event Data Structure Members . . . . .         | 3-6   |
| 3-7  | Circulate Request Event Data Structure Members . . . . . | 3-7   |
| 3-8  | Class Hint Data Structure Members . . . . .              | 3-7   |
| 3-9  | Client Message Event Data Structure Members . . . . .    | 3-8   |
| 3-10 | Color Data Structure Members . . . . .                   | 3-9   |
| 3-11 | Color Map Event Data Structure Members . . . . .         | 3-10  |
| 3-12 | Compose Data Structure Members . . . . .                 | 3-10  |
| 3-13 | Configure Event Data Structure Members . . . . .         | 3-11  |
| 3-14 | Configure Request Event Data Structure Members . . . . . | 3-13  |
| 3-15 | Create Window Event Data Structure Members . . . . .     | 3-14  |
| 3-16 | Crossing Event Data Structure Members . . . . .          | 3-16  |
| 3-17 | Destroy Window Event Data Structure Members . . . . .    | 3-17  |
| 3-18 | Error Event Data Structure Members . . . . .             | 3-18  |
| 3-19 | Event Mask Elements . . . . .                            | 3-19  |
| 3-20 | Expose Event Data Structure Members . . . . .            | 3-20  |
| 3-21 | Ext Codes Data Structure . . . . .                       | 3-21  |
| 3-22 | Ext Data Data Structure . . . . .                        | 3-22  |
| 3-23 | Focus Change Event Data Structure Members . . . . .      | 3-22  |
| 3-24 | Font Property Data Structure Members . . . . .           | 3-23  |
| 3-25 | Font Data Structure Members . . . . .                    | 3-25  |
| 3-26 | GC Values Data Structure Members . . . . .               | 3-27  |
| 3-27 | GC Mask Bits . . . . .                                   | 3-28  |
| 3-28 | Graphics Expose Event Data Structure Members . . . . .   | 3-30  |
| 3-29 | Gravity Event Data Structure Members . . . . .           | 3-31  |
| 3-30 | Host Address Data Structure Members . . . . .            | 3-32  |
| 3-31 | Icon Size Data Structure Members . . . . .               | 3-32  |
| 3-32 | Image Data Structure Members . . . . .                   | 3-34  |
| 3-33 | Key Event Data Structure Members . . . . .               | 3-35  |
| 3-34 | Keyboard Control Data Structure Members . . . . .        | 3-37  |
| 3-35 | Keyboard State Data Structure Members . . . . .          | 3-37  |
| 3-36 | Keymap Event Data Structure Members . . . . .            | 3-38  |
| 3-37 | Map Event Data Structure Members . . . . .               | 3-39  |
| 3-38 | Map Request Event Data Structure Members . . . . .       | 3-40  |

|      |                                                      |      |
|------|------------------------------------------------------|------|
| 3-39 | Mapping Event Data Structure Members .....           | 3-41 |
| 3-40 | Modifier Keymap Data Structure Members .....         | 3-42 |
| 3-41 | Motion Event Data Structure Members .....            | 3-43 |
| 3-42 | No Expose Event Data Structure Members .....         | 3-44 |
| 3-43 | Pixmap Format Values Data Structure Members .....    | 3-44 |
| 3-44 | Point Data Structure Members .....                   | 3-45 |
| 3-45 | Property Event Data Structure Members .....          | 3-46 |
| 3-46 | Rectangle Data Structure Members .....               | 3-46 |
| 3-47 | Reparent Event Data Structure Members .....          | 3-47 |
| 3-48 | Resize Request Event Data Structure Members .....    | 3-48 |
| 3-49 | Resource Manager Option Data Structure Members ..... | 3-49 |
| 3-50 | Resource Manager Value Data Structure Members .....  | 3-50 |
| 3-51 | Segment Data Structure Members .....                 | 3-50 |
| 3-52 | Selection Clear Event Data Structure Members .....   | 3-51 |
| 3-53 | Selection Event Data Structure Members .....         | 3-52 |
| 3-54 | Selection Request Event Data Structure Members ..... | 3-53 |
| 3-55 | Set Window Attributes Data Structure Members .....   | 3-54 |
| 3-56 | Size Hints Data Structure Members .....              | 3-56 |
| 3-57 | Standard Color Map Data Structure Members .....      | 3-57 |
| 3-58 | Text Item Data Structure Members .....               | 3-58 |
| 3-59 | Text Item 16 Data Structure Members .....            | 3-59 |
| 3-60 | Text Property Data Structure Members .....           | 3-59 |
| 3-61 | Time Coordinate Data Structure Members .....         | 3-60 |
| 3-62 | Unmap Event Data Structure Members .....             | 3-61 |
| 3-63 | Visibility Event Data Structure Members .....        | 3-61 |
| 3-64 | Visual Info Data Structure Members .....             | 3-62 |
| 3-65 | Window Attributes Data Structure Members .....       | 3-64 |
| 3-66 | Window Changes Data Structure Members .....          | 3-66 |
| 3-67 | WM Hints Data Structure Members .....                | 3-67 |
| 3-68 | Xlib Error Codes .....                               | 3-68 |

## Intended Audience

This manual is intended for programmers who develop DECwindows Motif applications in languages other than C. Use this guide in conjunction with the following reference manuals:

- *X Window System*, third edition, provides detailed descriptions of each Xlib routine, as well as the Inter-Client Communication Conventions Manual (ICCCM), the X Logical Font Description Conventions, and the X Window System Protocol.
- *X Window System Toolkit* provides reference information on the Intrinsics.
- *OSF/Motif Programmer's Reference* provides reference information on the Motif Toolkit.
- *DECwindows Extensions to Motif* provides reference information on the Digital extensions to Motif.

## Document Structure

The bindings are presented in alphabetical order, followed by the associated routines in a separate chapter. This guide is divided into four parts and contains the following chapters:

Chapter 1—Introduction

Part I contains the following:

Chapter 2—Xlib Routines

Chapter 3—Xlib Data Structures and Error Codes

Part II contains the following:

Chapter 4—Intrinsics Routines

Chapter 5—Intrinsics Data Structures

Part III contains the following:

Chapter 6—OSF/Motif Toolkit Routines

Chapter 7—OSF/Motif Toolkit Data Structures

Part IV contains the following:

Chapter 8—DECwindows Toolkit Extensions Routines

Chapter 9—DECwindows Toolkit Extensions Data Structures

## Associated Documents

In addition to the reference information, see also:

- *X and Motif Quick Reference Guide* provides quick reference information on Xlib, Intrinsics, and the Motif Toolkit.
- *DECwindows Motif Guide to Application Programming* describes how to program with the Digital extensions to the Motif Toolkit. It supplements the *OSF/Motif Programmer's Guide*.
- *Porting XUI Applications to Motif* describes how to port an existing XUI DECwindows application to Motif.
- *VMS DECwindows Guide to Xlib (Release 4) Programming: MIT C Binding* describes how to program with Xlib using MIT C bindings.
- *VMS DECwindows Guide to Xlib (Release 4) Programming: VAX Binding* describes how to program with Xlib using VAX bindings.
- *OSF/Motif Style Guide* describes style guidelines for applications based on the Motif Toolkit.
- *OSF/Motif Programmer's Guide* describes how to program with the Motif Window Manager, Motif Toolkit, and the Motif User Interface Language (UIL).

## Conventions

The following conventions are used in this manual:

In this manual, every use of DECwindows and DECwindows Motif refers to DECwindows Motif for OpenVMS software.

In this manual, every use of OpenVMS AXP means the OpenVMS AXP operating system, every use of OpenVMS VAX means the OpenVMS VAX operating system, and every use of OpenVMS means both the OpenVMS AXP operating system and the OpenVMS VAX operating system.

Ctrl/*x*                      A sequence such as Ctrl/*x* indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.

PF1 *x*                        A sequence such as PF1 *x* indicates that you must first press and release the key labeled PF1 and then press and release another key or a pointing device button.

GOLD *x*                      A sequence such as GOLD *x* indicates that you must first press and release the key defined as GOLD and then press and release another key. GOLD key sequences can also have a slash (/), dash (-), or underscore (\_) as a delimiter in EVE commands.

The GOLD key definition is often mapped to the PF1 key on the keypad.

Return                      In examples, a key name enclosed in a box indicates that you press a key on the keyboard. (In text, a key name is not enclosed in a box.)

|                      |                                                                                                                                                                                                                                                                                                                                                                  |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ...                  | Horizontal ellipsis points in examples indicate one of the following possibilities: <ul style="list-style-type: none"> <li>• Additional optional arguments in a statement have been omitted.</li> <li>• The preceding item or items can be repeated one or more times.</li> <li>• Additional parameters, values, or other information can be entered.</li> </ul> |
| .<br>. .<br>. . .    | Vertical ellipsis points indicate the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.                                                                                                                                                                                |
| ( )                  | In command format descriptions, parentheses indicate that, if you choose more than one option, you must enclose the choices in parentheses.                                                                                                                                                                                                                      |
| [ ]                  | In command format descriptions, brackets indicate optional elements. You can choose one, none, or all of the options. (Brackets are not optional, however, in the syntax of a directory name in an OpenVMS file specification or in the syntax of a substring specification in an assignment statement.)                                                         |
| { }                  | In command format descriptions, braces surround a required choice of options; you must choose one of the options listed.                                                                                                                                                                                                                                         |
| <b>boldface text</b> | Boldface text represents the introduction of a new term or the name of an argument, an attribute, or a reason (user action that triggers a callback).<br><br>Boldface text is also used to show user input in Bookreader versions of the manual.                                                                                                                 |
| <i>italic text</i>   | Italic text emphasizes important information and indicates complete titles of manuals and variables. Variables include information that varies in system messages (Internal error <i>number</i> ), in command lines ( <i>/PRODUCER=name</i> ), and in command parameters in text (where <i>device-name</i> contains up to five alphanumeric characters).         |
| UPPERCASE TEXT       | Uppercase text indicates a command, the name of a routine, the name of a file, or the abbreviation for a system privilege.                                                                                                                                                                                                                                       |
| struct               | Monospace type in text identifies the following C programming language elements: keywords, the names of independently compiled external functions and files, syntax summaries, and references to variables or identifiers introduced in an example.                                                                                                              |
| -                    | A hyphen in code examples indicates that additional arguments to the request are provided on the line that follows.                                                                                                                                                                                                                                              |
| numbers              | All numbers in text are assumed to be decimal unless otherwise noted. Nondecimal radices—binary, octal, or hexadecimal—are explicitly indicated.                                                                                                                                                                                                                 |
| mouse                | The term <i>mouse</i> refers to any pointing device, such as a mouse, a puck, or a stylus.                                                                                                                                                                                                                                                                       |
| MB1, MB2, MB3        | MB1 indicates the left mouse button, MB2 indicates the middle mouse button, and MB3 indicates the right mouse button. (The user can redefine the buttons.)                                                                                                                                                                                                       |
| PB1, PB2, PB3, PB4   | PB1, PB2, PB3, and PB4 indicate buttons on the puck.                                                                                                                                                                                                                                                                                                             |
| SB1, SB2, SB3        | SB1, SB2, and SB3 indicate buttons on the stylus.                                                                                                                                                                                                                                                                                                                |



---

# Introduction

This manual provides OpenVMS documentation required by DECwindows programmers using languages other than C. This chapter outlines each section and element in the routine templates.

There are two separate routine templates, one for each binding format:

- Standard OpenVMS bindings are supplied for Xlib routines; the template is described in Section 1.1.
- The Intrinsics, OSF/Motif Toolkit, and Digital Extension Routines are in generic form; the language bindings files supply the language-specific form of each entry. The template for these routines is described in Section 1.2.

## 1.1 Xlib Routine Template

This section describes each element in the Xlib Routine template.

### Routine Name

The generic name of the routine.

### OpenVMS Format

The OpenVMS routine call format. The format syntax is summarized in Table 1–1.

**Table 1–1 General Rules of Syntax**

| Element                  | Syntax Rule                                                                                                                                                                                                                                            |
|--------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Routine entry point name | The name is shown in all capital letters with the prefix X. The routine entry point name is required.                                                                                                                                                  |
| Equal sign               | For specific return values, the equal sign is required.                                                                                                                                                                                                |
| Parentheses              | Open and close parentheses surround the argument list in a routine call. Parentheses are required.                                                                                                                                                     |
| Argument names           | Argument names, including names of return arguments, are always shown in lowercase characters. All arguments not enclosed by brackets ([]) are required. Argument names must be listed in the same order in your program as they appear in the format. |

(continued on next page)

## Introduction

### 1.1 Xlib Routine Template

**Table 1–1 (Cont.) General Rules of Syntax**

| Element  | Syntax Rule                                                                                                                                                               |
|----------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Spaces   | A line break follows the entry point name and the arguments. A comma, followed by a space separate each argument. Spaces are not required.                                |
| Brackets | Brackets ([]) surround optional arguments. Commas that appear inside brackets are optional and appear only when the optional argument appears. Brackets are not required. |
| Commas   | Commas must appear between required arguments. When commas appear inside brackets ([ ]), they appear only when the optional argument appears.                             |

#### Argument Information

A table follows the OpenVMS format listing. The table lists, in order, each argument name, usage, data type, access, and mechanism.

#### Usage

The Usage column specifies the general OpenVMS binding argument type. For example, if the argument is a resource identifier, the Usage field is “identifier.” This field provides additional information about the argument that is helpful when declaring the argument within a program. See Table 1–2 for a list of usage entries used in Xlib routines.

**Table 1–2 OpenVMS Usage Entries**

| Entry        | Description                                                                                                                                                                                                                     |
|--------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| address      | An unsigned longword containing the virtual address of data or code, but not of a procedure entry mask (which is a procedure entry).                                                                                            |
| any          | An unsigned longword containing either data or a pointer to data.                                                                                                                                                               |
| array        | An array with the specific description of its elements provided in the argument description.                                                                                                                                    |
| Boolean      | An unsigned longword with the predefined values of 1 for true and 0 for false.                                                                                                                                                  |
| byte         | A signed byte integer.                                                                                                                                                                                                          |
| uns byte     | An unsigned byte integer.                                                                                                                                                                                                       |
| char string  | A string of 0 to 65,535 8-bit characters.                                                                                                                                                                                       |
| cond value   | Unsigned longword specifying that a predefined condition value will be returned in R0.<br>The cond value status return is SSS_NORMAL for success; the return value can be either zero or an OpenVMS condition code for failure. |
| identifier   | A value used to refer to a resource. It is originally returned by the system. In the OpenVMS binding, an identifier is a longword integer (unsigned).                                                                           |
| longword     | A signed longword integer.                                                                                                                                                                                                      |
| uns longword | An unsigned longword integer.                                                                                                                                                                                                   |

(continued on next page)



**Table 1–2 (Cont.) OpenVMS Usage Entries**

| Entry         | Description                                                                                           |
|---------------|-------------------------------------------------------------------------------------------------------|
| mask longword | An unsigned longword interpreted as a bit mask.                                                       |
| procedure     | An entry mask to a procedure.                                                                         |
| record        | A data structure with the specific description of the structure provided in the argument description. |
| word          | A signed word integer.                                                                                |
| uns word      | An unsigned word integer.                                                                             |

**Data Type**

The Data Type column specifies the standard OpenVMS data type of the argument. For example, if the argument is a resource identifier, the data type is “longword (unsigned).” If the argument has a predefined value, it is provided in the description of the argument.

The following OpenVMS data type entries are used in the Xlib routines:

- byte
- character string
- longword
- longword (unsigned), notated as *uns longword*
- proc entry mask
- vector (unsigned) longword, notated as *v uns longword*
- vector (unsigned) byte, notated as *v uns byte*
- word (signed)
- word (unsigned), notated as *uns word*
- Name of an OpenVMS data structure. These are valid OpenVMS data structure names that can appear in this column (see Chapter 3 for more information about OpenVMS data structures):

|                       |                      |
|-----------------------|----------------------|
| x\$any_event          | x\$keyboard_control  |
| x\$arc                | x\$keyboard_state    |
| x\$button_event       | x\$keymap_event      |
| x\$char_2b            | x\$map_event         |
| x\$char_struct        | x\$map_request_event |
| x\$circ_request_event | x\$mapping_event     |
| x\$circulate_event    | x\$modifier_keymap   |
| x\$class_hint         | x\$motion_event      |
| x\$clie_message_event | x\$no_expose_event   |
| x\$color              | x\$point             |
| x\$colormap_event     | x\$property_event    |
| x\$compose_status     | x\$rectangle         |
| x\$conf_request_event | x\$parent_event      |

## Introduction

### 1.1 Xlib Routine Template

|                       |                       |
|-----------------------|-----------------------|
| x\$configure_event    | x\$resource_req       |
| x\$creat_window_event | x\$resz_request_event |
| x\$crossing_event     | x\$rm_value           |
| x\$depth              | x\$screen             |
| x\$destr_window_event | x\$screen_format      |
| x\$display            | x\$segment            |
| x\$error_event        | x\$sel_request_event  |
| x\$event              | x\$select_clear_event |
| x\$expose_event       | x\$selection_event    |
| x\$ext_codes          | x\$set_win_attributes |
| x\$ext_data           | x\$size_hints         |
| x\$focus_change_event | x\$standard_colormap  |
| x\$font_prop          | x\$text_item          |
| x\$font_struct        | x\$text_item_16       |
| x\$frame              | x\$text_property      |
| x\$gc_struct          | x\$time_coord         |
| x\$gc_values          | x\$unmap_event        |
| x\$graph_expose_event | x\$visibility_event   |
| x\$gravity_event      | x\$visual             |
| x\$host_address       | x\$visual_info        |
| x\$icon_size          | x>window_attributes   |
| x\$image              | x>window_changes      |
| x\$key_event          | x\$wm_hints           |

#### Access

The Access column specifies the way in which the called routine accesses the argument. For example, when the argument is passed as input, the access is “read only” for both bindings and when the argument is returned by the routine, the access is “write only” for both bindings. See Table 1–3 for a list of access entries used in the Xlib routines.

**Table 1–3 Access Entries**

| Entry  | Description                                                                                                                                                                                                                                             |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| read   | Input data required by the routine to perform its operation must be readable. When an argument specifies input data, the access entry is readable. The routine cannot write data back to this argument.                                                 |
| write  | Output data returned by the routine to a specific location. When an argument specifies output data, the access entry is writable. The routine does not read the contents of the location either before or after it writes into the location.            |
| modify | The routine reads the input data, which it uses in its operation, and then overwrites the input data with the results (the output data) of the operation. Thus, when the routine completes execution, the input data specified by the argument is lost. |

**Mechanism**

The Mechanism column specifies the passing mechanism used by the called routine. For example, when the argument is the value itself, the mechanism is “by value”; when the argument is a pointer to the value, the mechanism is “by reference.” See Table 1–4 for a complete list of mechanism entries used in the Xlib routines.

**Table 1–4 Mechanism Entries**

| Entry      | Description                                                                                                                                                                                                                                                                                                                                                 |
|------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| value      | The argument contains the actual data to be used by the routine. Note that because an argument is only one longword in length, only data that can be represented in one longword can be passed by value.                                                                                                                                                    |
| reference  | The argument contains the address of the data to be used by the routine. The argument is a pointer to the actual data.                                                                                                                                                                                                                                      |
| descriptor | The argument contains the address of a descriptor. A descriptor consists of two or more longwords (depending on the type of descriptor used), which describes the location, length, and the OpenVMS standard data type of the data to be used by the called routine. The argument is a pointer to a descriptor that itself is a pointer to the actual data. |

## 1.2 Intrinsic, OSF/Motif Toolkit, and Digital Extension Routines Template

This section describes each element in the template that documents the Intrinsic, OSF/Motif Toolkit, and Digital Extension routines.

**Routine Name**

The generic name of the routine.

**Format**

The generic routine call format. The bindings file for each language supplies the correct elements for that language. Table 1–5 describes the syntax conventions.

**Table 1–5 Syntax Conventions**

| Element                  | Conventions                                                                                                                                                                                                |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Routine entry point name | The name is shown in its generic form, mixed case, with the prefixes Xt, Xm or DXm corresponding to Intrinsic, OSF/Motif Toolkit, or Digital Extension Routines. The routine entry point name is required. |
| Return values            | If there is a return value, it precedes the entry point name and is in the same font as the arguments. A void is implied when there is no return value.                                                    |
| Parentheses              | Open and close parentheses surround the argument list in a routine call. Parentheses are required.                                                                                                         |

(continued on next page)

## Introduction

### 1.2 Intrinsic, OSF/Motif Toolkit, and Digital Extension Routines Template

**Table 1–5 (Cont.) Syntax Conventions**

| Element        | Conventions                                                                                                                                                                             |
|----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Argument names | Argument names, including names of return arguments, appear in mixed case. Arguments in brackets are optional. In a program, list the arguments in the order they appear in the format. |
| Spaces         | One or more spaces are shown between the entry point name and the first argument, and between each argument. Spaces are not required.                                                   |
| Brackets       | Brackets ([]) surround optional arguments. Commas that appear inside brackets are optional and appear only when the optional argument appears. Brackets are not required.               |
| Commas         | Commas must appear between required arguments. When commas appear inside brackets ([]), they appear only when the optional argument appears.                                            |

A table containing information about each argument follows the format section. Arguments are described in order, beginning with the return value, if any. The table lists each argument name, type, access, and mechanism.

#### Type

Table 1–6 describes the argument type entries. Type names in all caps are generic types; the bindings files for each language translate them into the conventions of that language. Types in mixed case are defined within the language bindings files and can, in most cases, be used as documented.

**Table 1–6 Type Entries**

| Entry         | Description                                                                                                                                                                                                                                                                          |
|---------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| ADDRESS       | An unsigned longword containing the virtual address of data or code.                                                                                                                                                                                                                 |
| ANY           | An unsigned longword containing either data or a pointer to data.                                                                                                                                                                                                                    |
| Arg           | A structure consisting of two fields: name (the address of a character string) and value (a longword integer containing either data or a pointer to data).                                                                                                                           |
| ArgList       | An array of Arg structures.                                                                                                                                                                                                                                                          |
| Atom          | An unsigned longword integer interpreted as an X atom.                                                                                                                                                                                                                               |
| Boolean       | A single byte with the predefined values of 1 for true and 0 for false. The language bindings define the constants XtTrue as one and XtFalse as zero for the purpose of setting Boolean values in languages where the native values of true and false do not equate to one and zero. |
| BYTE          | A signed byte integer.                                                                                                                                                                                                                                                               |
| BYTE UNSIGNED | An unsigned byte integer.                                                                                                                                                                                                                                                            |
| Cardinal      | An unsigned longword integer.                                                                                                                                                                                                                                                        |

(continued on next page)

## 1.2 Intrinsic, OSF/Motif Toolkit, and Digital Extension Routines Template

Table 1–6 (Cont.) Type Entries

| Entry             | Description                                                                                                |
|-------------------|------------------------------------------------------------------------------------------------------------|
| CHARACTER         | A byte integer containing an ASCII character value.                                                        |
| Colormap          | An unsigned longword integer interpreted as an X color map.                                                |
| Cursor            | An unsigned longword integer interpreted as an X cursor.                                                   |
| Dimension         | An unsigned word integer.                                                                                  |
| Display           | An unsigned longword integer interpreted as an X display.                                                  |
| DXmColorMixWidget | A DXm color mix widget.                                                                                    |
| DXmCSTextStatus   | An unsigned longword integer (actually a C enumerated type).                                               |
| DXmCSTextPosition | A signed longword integer.                                                                                 |
| DxmCSTextWidget   | A DXm compound string text widget.                                                                         |
| Event             | An X event structure.                                                                                      |
| EventMask         | An unsigned longword integer interpreted as an X event mask.                                               |
| Font              | An unsigned longword integer interpreted as an X font.                                                     |
| GC                | An unsigned longword integer interpreted as an X graphics context.                                         |
| I18nContext       | Address of an I18nContextRec structure.                                                                    |
| KeyCode           | An unsigned longword integer interpreted as an X keycode.                                                  |
| KeySym            | An unsigned longword integer interpreted as an X keysym.                                                   |
| LONGWORD          | A signed longword integer.                                                                                 |
| LONGWORD UNSIGNED | An unsigned longword integer.                                                                              |
| Modifiers         | An unsigned longword integer interpreted as an Intrinsic modifier key mask.                                |
| MrmCode           | A signed word integer.                                                                                     |
| MrmCount          | A signed word integer.                                                                                     |
| MrmHierarchy      | An address interpreted as an Mrm hierarchy pointer.                                                        |
| MrmOsOpenParamPtr | The address of an MrmOsOpenParam structure.                                                                |
| MrmRegisterArg    | A structure consisting of two fields: name (the address of a character string) and value (of type Opaque). |
| MrmType           | A signed word integer.                                                                                     |
| Opaque            | A value of type address.                                                                                   |
| Pixel             | An unsigned longword integer interpreted as an X pixel.                                                    |
| Pixmap            | An unsigned longword integer interpreted as an X pixmap.                                                   |
| Position          | A signed word integer.                                                                                     |

(continued on next page)

## Introduction

### 1.2 Ininsics, OSF/Motif Toolkit, and Digital Extension Routines Template

Table 1–6 (Cont.) Type Entries

| Entry                  | Description                                                                   |
|------------------------|-------------------------------------------------------------------------------|
| Region                 | An unsigned longword integer interpreted as an X region.                      |
| Screen                 | An unsigned longword integer interpreted as an X screen.                      |
| Substitution           | The address of an Intrinsic Substitution Rec.                                 |
| Time                   | An unsigned longword integer interpreted as an X timestamp value.             |
| VoidProc               | An entry mask to a procedure with no return value.                            |
| Widget                 | An address interpreted as a widget identifier.                                |
| WidgetClass            | An address interpreted as a widget class pointer.                             |
| WidgetList             | An array of widgets.                                                          |
| Window                 | An unsigned longword integer interpreted as an X window.                      |
| WORD UNSIGNED          | An unsigned word integer.                                                     |
| XEvent                 | An Xlib event structure.                                                      |
| XImage                 | An Xlib XImage structure.                                                     |
| XGCValues              | An unsigned longword interpreted as a pointer to an Xlib XGCValues structure. |
| XKeyPressedEvent       | An X event.                                                                   |
| XmClipboardPendingList | An array of XmClipboardPendingRec structures.                                 |
| XmColorProc            | An entry mask to a procedure with no return value.                            |
| XmCutPasteProc         | A callback procedure.                                                         |
| XmDropSiteVisuals      | An XmDropSiteVisualsRec structure type.                                       |
| XmDropTransferEntry    | An XmDropTransferEntryRec structure type.                                     |
| XmFontContext          | An address interpreted as a Motif font context.                               |
| XmFontList             | An address interpreted as a Motif font list.                                  |
| XmFontListEntry        | An XmFontListRec structure type.                                              |
| XmFontType             | An unsigned longword.                                                         |
| XmHighlightMode        | An unsigned longword (actually a C enumerated type).                          |
| XmOffset               | A signed longword integer.                                                    |
| XmOffsetPtr            | The address of a variable of type XmOffset.                                   |
| XmRepTypeId            | An unsigned word.                                                             |
| XmRepTypeList          | An XmRepTypeListRec structure type.                                           |
| XmString               | An address interpreted as a Motif compound string.                            |
| XmStringCharSet        | An address interpreted as a Motif character set identifier.                   |
| XmStringComponentType  | An unsigned byte integer.                                                     |
| XmStringContext        | An address interpreted as a Motif compound string context.                    |
| XmStringDirection      | An unsigned byte integer.                                                     |

(continued on next page)

## 1.2 Intrinsic, OSF/Motif Toolkit, and Digital Extension Routines Template

Table 1–6 (Cont.) Type Entries

| Entry                        | Description                                                       |
|------------------------------|-------------------------------------------------------------------|
| XmStringTable                | An array of XmStrings.                                            |
| XmStringTextBlock            | The address of an XmTextBlockRec.                                 |
| XmStringTextFormat           | An unsigned longword integer.                                     |
| XmTextBlock                  | A structure used to pass text.                                    |
| XmTextDirection              | An unsigned longword.                                             |
| XmTextPosition               | A signed longword integer.                                        |
| XmTextSource                 | An address interpreted as a Motif text source.                    |
| XmTraversalDirection         | An unsigned longword that describes traversal direction.          |
| XmVisibility                 | An unsigned longword.                                             |
| XRectangle                   | An X rectangle structure.                                         |
| XrmDatabase                  | An unsigned longword integer.                                     |
| XrmValuePtr                  | An address interpreted as an Xrm value pointer.                   |
| XtAddressMode                | An unsigned longword integer (actually a C enumerated type).      |
| XtAccelerators               | An address interpreted as a compiled Intrinsic accelerator table. |
| XtActionHookId               | A value of Opaque type.                                           |
| XtActionHookProc             | An entry mask to a procedure with no return value.                |
| XtActionList                 | An array of Intrinsic XtActionsRec structures.                    |
| XtActionProc                 | An entry mask to a procedure with no return value.                |
| XtAppContext                 | An address interpreted as an Intrinsic application context.       |
| XtArgVal                     | A signed longword integer.                                        |
| XtCallbackList               | An array of Intrinsic XtCallbackRec structures.                   |
| XtCallbackProc               | An entry mask to a procedure with no return value.                |
| XtCacheRef                   | A value of Opaque type.                                           |
| XtCacheType                  | A signed longword integer.                                        |
| XtCancelConvertSelectionProc | An entry mask to a procedure with no return value.                |
| XtCaseProc                   | An entry mask to a procedure with no return value.                |
| XtConvertSelectionProc       | An entry mask to a procedure with a Boolean return value.         |
| XtConverter                  | An entry mask to a procedure with no return value.                |
| XtDestructor                 | An entry mask to a procedure with no return value.                |
| XtErrorHandler               | An entry mask to a procedure with no return value.                |
| XtErrorMsgHandler            | An entry mask to a procedure with no return value.                |
| XtEventHandler               | An entry mask to a procedure with no return value.                |
| XtFilePredicate              | An entry mask to a procedure with a Boolean return value.         |
| XtGCmask                     | An unsigned longword integer.                                     |

(continued on next page)

## Introduction

### 1.2 Ininsics, OSF/Motif Toolkit, and Digital Extension Routines Template

**Table 1–6 (Cont.) Type Entries**

| Entry                   | Description                                                         |
|-------------------------|---------------------------------------------------------------------|
| XtGeometryMask          | An unsigned longword integer.                                       |
| XtGeometryResult        | An unsigned longword integer (actually a C enumerated type).        |
| XtGrabKind              | An unsigned longword integer (actually a C enumerated type).        |
| XtIntervalId            | An unsigned longword integer.                                       |
| XtInputCallbackProc     | An entry mask to a procedure with no return value.                  |
| XtInputId               | An unsigned longword integer.                                       |
| XtInputMask             | An unsigned longword integer.                                       |
| XtLanguageProc          | An entry mask to a procedure that returns a null-terminated string. |
| XtListPosition          | An unsigned longword integer.                                       |
| XtLoseSelectionProc     | An entry mask to a procedure with no return value.                  |
| XtPointer               | A value of type address.                                            |
| XtSelectionCallbackProc | An entry mask to a procedure with no return value.                  |
| XtSelectionDoneProc     | An entry mask to a procedure with no return value.                  |
| XtString                | A null-terminated string of from 0 to 65,535 8-bit characters.      |
| XtStringArray           | An array of XtStrings.                                              |
| XtTimerCallbackProc     | An entry mask to a procedure with no return value.                  |
| XtTranslations          | An address interpreted as a compiled Intrinsic translation table.   |
| XtTypeConverter         | An entry mask to a procedure with a Boolean return value.           |
| XtWidgetGeometry        | An XtWidgetGeometry structure.                                      |
| XtWorkProc              | An entry mask to a procedure with a Boolean return value.           |
| XtWorkProcID            | An unsigned longword integer.                                       |

#### Access

The Access column specifies the way in which the called routine accesses the argument. For example, when the argument is passed as input, the access is “read only” and when the argument is returned by the routine, the access is “write only.” See Table 1–7 for a list of access entries used.

**Table 1–7 Access Entries**

| Entry | Description                                                                                                                                                                                             |
|-------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| read  | Input data required by the routine to perform its operation must be readable. When an argument specifies input data, the access entry is readable. The routine cannot write data back to this argument. |

(continued on next page)



## 1.2 Intrinsic, OSF/Motif Toolkit, and Digital Extension Routines Template

Table 1–7 (Cont.) Access Entries

| Entry  | Description                                                                                                                                                                                                                                             |
|--------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| write  | Output data returned by the routine to a specific location. When an argument specifies output data, the access entry is writable. The routine does not read the contents of the location either before or after it writes into the location.            |
| modify | The routine reads the input data, which it uses in its operation, and then overwrites the input data with the results (the output data) of the operation. Thus, when the routine completes execution, the input data specified by the argument is lost. |

**Mechanism**

The Mechanism column specifies the passing mechanism used by the called routine. For example, when the argument is the value itself, the mechanism is “by value” and when the argument is a pointer to the value, the mechanism is “by reference.” See Table 1–8 for a complete list of mechanism entries.

Table 1–8 Mechanism Entries

| Entry     | Description                                                                                                            |
|-----------|------------------------------------------------------------------------------------------------------------------------|
| value     | The argument contains the actual data to be used by the routine.                                                       |
| reference | The argument contains the address of the data to be used by the routine. The argument is a pointer to the actual data. |



# Part I

---

## DECwindows Motif for OpenVMS Xlib

Part I documents Xlib routines and data structures and includes the following chapters:

- Chapter 2 – Xlib Routines
- Chapter 3 – Xlib Data Structures and Error Codes



---

## Xlib Routines

This chapter lists Xlib routines. Some bindings are provided for the Release 5 Xlib routines. See Section 1.1 for information about the format used to describe each routine.

Refer to the corresponding C routines in the *X Window System* for additional information about the description of each function and a discussion of related background information.

---

### ACTIVATE\_SCREEN\_SAVER

#### OpenVMS Format

X\$ACTIVATE\_SCREEN\_SAVER  
(display)

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

### ADD\_HOST

#### OpenVMS Format

X\$ADD\_HOST  
(display, host)

#### Argument Information

| Argument | Usage      | Data Type       | Access | Mechanism |
|----------|------------|-----------------|--------|-----------|
| display  | identifier | uns longword    | read   | reference |
| host     | record     | x\$host_address | read   | reference |

## Xlib Routines

### ADD HOSTS

---

## ADD HOSTS

### OpenVMS Format

X\$ADD\_HOSTS  
(display, hosts, num\_hosts)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| hosts     | array      | uns longword | read   | reference |
| num_hosts | longword   | uns longword | read   | reference |

---

## ADD PIXEL

### OpenVMS Format

X\$ADD\_PIXEL  
(ximage, value)

### Argument Information

| Argument | Usage    | Data Type | Access | Mechanism |
|----------|----------|-----------|--------|-----------|
| ximage   | record   | x\$image  | read   | reference |
| value    | longword | longword  | read   | reference |

---

## ADD TO SAVE SET

### OpenVMS Format

X\$ADD\_TO\_SAVE\_SET  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## ALLOCCOLOR

### OpenVMS Format

X\$ALLOCCOLOR  
(class\_hints\_return)

### Argument Information

| Argument           | Usage  | Data Type     | Access | Mechanism |
|--------------------|--------|---------------|--------|-----------|
| class_hints_return | record | x\$class_hint | write  | reference |

---

## ALLOCCOLOR

### OpenVMS Format

status\_return = X\$ALLOCCOLOR  
(display, colormap\_id, screen\_in\_out)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| colormap_id   | identifier | uns longword | read   | reference |
| screen_in_out | record     | x\$color     | modify | reference |

### Returns

**status\_return**  
ALLOCCOLOR returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| SSS_NORMAL     | Routine completed successfully. |
| X\$_ERRORREPLY | Error received from server.     |

---

## ALLOC COLOR CELLS

### OpenVMS Format

status\_return = X\$ALLOC\_COLOR\_CELLS

(display, colormap\_id, contig, plane\_masks\_return, num\_planes, pixels\_return,  
num\_colors)

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| status_return      | cond_value | uns longword | write  | value     |
| display            | identifier | uns longword | read   | reference |
| colormap_id        | identifier | uns longword | read   | reference |
| contig             | Boolean    | uns longword | read   | reference |
| plane_masks_return | array      | uns longword | write  | reference |
| num_planes         | longword   | uns longword | read   | reference |
| pixels_return      | array      | uns longword | write  | reference |
| num_colors         | longword   | uns longword | read   | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. ALLOC COLOR CELLS returns one of the following values:

| Value         | Description                     |
|---------------|---------------------------------|
| SS\$NORMAL    | Routine completed successfully. |
| X\$ERRORREPLY | Error received from server.     |

---

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **num\_planes**, **num\_colors**.

---



---

## ALLOC COLOR PLANES

### OpenVMS Format

status\_return = X\$ALLOC\_COLOR\_PLANES

(display, colormap\_id, contig, pixels\_return, num\_colors, num\_reds, num\_greens,  
num\_blues, rmask\_return, gmask\_return, bmask\_return)

### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| status_return | cond_value    | uns longword | write  | value     |
| display       | identifier    | uns longword | read   | reference |
| colormap_id   | identifier    | uns longword | read   | reference |
| contig        | Boolean       | uns longword | read   | reference |
| pixels_return | array         | uns longword | write  | reference |
| num_colors    | longword      | longword     | read   | reference |
| num_reds      | longword      | longword     | read   | reference |
| num_greens    | longword      | longword     | read   | reference |
| num_blues     | longword      | longword     | read   | reference |
| rmask_return  | mask_longword | uns longword | write  | reference |
| gmask_return  | mask_longword | uns longword | write  | reference |
| bmask_return  | mask_longword | uns longword | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. ALLOC COLOR PLANES returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| SS\$_NORMAL    | Routine completed successfully. |
| X\$_ERRORREPLY | Error received from server.     |

---

## ALLOC ICON SIZE

### OpenVMS Format

X\$ALLOC\_ICON\_SIZE

(icon\_size\_return)

## Xlib Routines

### ALLOC ICON SIZE

#### Argument Information

| Argument         | Usage  | Data Type    | Access | Mechanism |
|------------------|--------|--------------|--------|-----------|
| icon_size_return | record | x\$icon_size | write  | reference |

---

## ALLOC NAMED COLOR

### OpenVMS Format

status\_return = X\$ALLOC\_NAMED\_COLOR

(display, colormap\_id, color\_name, [screen\_def\_return], [exact\_def\_return])

#### Argument Information

| Argument            | Usage       | Data Type        | Access | Mechanism  |
|---------------------|-------------|------------------|--------|------------|
| status_return       | longword    | longword         | write  | value      |
| display             | identifier  | uns longword     | read   | reference  |
| colormap_id         | identifier  | uns longword     | read   | reference  |
| color_name          | char_string | character string | read   | descriptor |
| color_screen_return | record      | x\$color         | write  | reference  |
| color_exact_return  | record      | x\$color         | write  | reference  |

### Arguments

#### color\_screen\_return

The color data structure that returns the color definition of the color actually used by the server. The color definition and the color index are returned in the color data structure. This argument is optional.

#### color\_exact\_return

The color data structure that defines the exact color as specified in the color database. The color definition is returned in the color definition data structure. This argument is optional.

---

## ALLOC SIZE HINTS

### OpenVMS Format

X\$ALLOC\_SIZE\_HINTS

(size\_hints\_return)

**Argument Information**

| Argument          | Usage  | Data Type     | Access | Mechanism |
|-------------------|--------|---------------|--------|-----------|
| size_hints_return | record | x\$size_hints | write  | reference |

## ALLOC STANDARD COLORMAP

**OpenVMS Format**

X\$ALLOC\_STANDARD\_COLORMAP  
(standard\_colormap\_return)

**Argument Information**

| Argument                 | Usage  | Data Type            | Access | Mechanism |
|--------------------------|--------|----------------------|--------|-----------|
| standard_colormap_return | record | x\$standard_colormap | write  | reference |

## ALLOW EVENTS

**OpenVMS Format**

X\$ALLOW\_EVENTS  
(display, event\_mode, time)

**Argument Information**

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| event_mode | longword   | longword     | read   | reference |
| time       | longword   | uns longword | read   | reference |

**Arguments**

**event\_mode**

The events to be released. The predefined values for **event\_mode** are as follows:

X\$C\_ASYNC\_POINTER  
X\$C\_SYNC\_POINTER  
X\$C\_REPLAY\_POINTER  
X\$C\_ASYNC\_KEYBOARD  
X\$C\_SYNC\_KEYBOARD  
X\$C\_REPLAY\_KEYBOARD

## Xlib Routines

### ALLOW EVENTS

X\$C\_SYNC\_BOTH  
X\$C\_ASYNC\_BOTH

---

## ALL PLANES

### OpenVMS Format

value\_return = X\$ALL\_PLANES

### Argument Information

| Argument     | Usage        | Data Type    | Access | Mechanism |
|--------------|--------------|--------------|--------|-----------|
| value_return | uns longword | uns longword | write  | value     |

---

## AUTO REPEAT OFF

### OpenVMS Format

X\$AUTO\_REPEAT\_OFF  
(display)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## AUTO REPEAT ON

### OpenVMS Format

X\$AUTO\_REPEAT\_ON  
(display)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## BELL

### OpenVMS Format

X\$BELL  
(display, percent)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| percent  | longword   | longword     | read   | reference |

---

## BITMAP BIT ORDER

### OpenVMS Format

order\_return = X\$BITMAP\_BIT\_ORDER  
(display)

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| order_return | longword   | longword     | write  | value     |
| display      | identifier | uns longword | read   | reference |

### Returns

#### order\_return

The value of the leftmost bit in the bitmap, as displayed on the screen. Valid values are shown in the following table:

| Value          | Description                            |
|----------------|----------------------------------------|
| X\$C_LSB_FIRST | Least significant bit displayed first. |
| X\$C_MSB_FIRST | Most significant bit displayed first.  |

## Xlib Routines

### BITMAP PAD

---

## BITMAP PAD

### OpenVMS Format

pad\_return = X\$BITMAP\_PAD  
(display)

### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| pad_return | longword   | longword     | write  | value     |
| display    | identifier | uns longword | read   | reference |

---

## BITMAP UNIT

### OpenVMS Format

size\_return = X\$BITMAP\_UNIT  
(display)

### Argument Information

| Argument    | Usage        | Data Type    | Access | Mechanism |
|-------------|--------------|--------------|--------|-----------|
| size_return | uns longword | uns longword | write  | value     |
| display     | identifier   | uns longword | read   | reference |

---

## BLACK PIXEL

### OpenVMS Format

color\_index\_return = X\$BLACK\_PIXEL  
(display, screen\_number)

### Argument Information

| Argument           | Usage        | Data Type    | Access | Mechanism |
|--------------------|--------------|--------------|--------|-----------|
| color_index_return | identifier   | uns longword | write  | value     |
| display            | identifier   | uns longword | read   | reference |
| screen_number      | uns longword | uns longword | read   | reference |

## BLACK PIXEL OF SCREEN

### OpenVMS Format

```
color_index_return = X$BLACK_PIXEL_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| color_index_return | identifier | uns longword | write  | value     |
| screen_id          | identifier | uns longword | read   | reference |

## CELLS OF SCREEN

### OpenVMS Format

```
cells_return = X$CELLS_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| cells_return | longword   | longword     | write  | value     |
| screen_id    | identifier | uns longword | read   | reference |

## CHANGE ACTIVE POINTER GRAB

### OpenVMS Format

```
X$CHANGE_ACTIVE_POINTER_GRAB
    (display, event_mask, cursor_id, time)
```

### Argument Information

| Argument   | Usage         | Data Type    | Access | Mechanism |
|------------|---------------|--------------|--------|-----------|
| display    | identifier    | uns longword | read   | reference |
| event_mask | mask_longword | uns longword | read   | reference |
| cursor_id  | identifier    | uns longword | read   | reference |

## Xlib Routines

### CHANGE ACTIVE POINTER GRAB

| Argument | Usage    | Data Type    | Access | Mechanism |
|----------|----------|--------------|--------|-----------|
| time     | longword | uns longword | read   | reference |

### Arguments

#### event\_mask

Specifies the pointer events to be reported to the client. The mask can be the inclusive OR of the event mask values listed in Table 2–1.

**Table 2–1 Event Mask Description**

| Bit | Predefined Value        | Description                          |
|-----|-------------------------|--------------------------------------|
| 2   | XSM_BUTTON_PRESS        | Pointer button down events wanted    |
| 3   | XSM_BUTTON_RELEASE      | Pointer button up events wanted      |
| 4   | XSM_ENTER_WINDOW        | Pointer window entry events wanted   |
| 5   | XSM_LEAVE_WINDOW        | Pointer window leave events wanted   |
| 6   | XSM_POINTER_MOTION      | Pointer motion events wanted         |
| 7   | XSM_POINTER_MOTION_HINT | Pointer motion hints wanted          |
| 8   | XSM_BUTTON1_MOTION      | Pointer motion while button 1 down   |
| 9   | XSM_BUTTON2_MOTION      | Pointer motion while button 2 down   |
| 10  | XSM_BUTTON3_MOTION      | Pointer motion while button 3 down   |
| 11  | XSM_BUTTON4_MOTION      | Pointer while button 4 down          |
| 12  | XSM_BUTTON5_MOTION      | Pointer motion while button 5 down   |
| 13  | XSM_BUTTON_MOTION       | Pointer motion while any button down |
| 14  | XSM_KEYMAP_STATE        | Any keyboard state change wanted     |

## CHANGE GC

### OpenVMS Format

X\$CHANGE\_GC

(display, gc\_id, gc\_mask, values\_struct)

### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| display       | identifier    | uns longword | read   | reference |
| gc_id         | identifier    | uns longword | read   | reference |
| gc_mask       | mask_longword | uns longword | read   | reference |
| values_struct | record        | x\$gc_values | read   | reference |



---

## CHANGE KEYBOARD CONTROL

### OpenVMS Format

X\$CHANGE\_KEYBOARD\_CONTROL  
(display, value\_mask, control\_values)

### Argument Information

| Argument       | Usage         | Data Type           | Access | Mechanism |
|----------------|---------------|---------------------|--------|-----------|
| display        | identifier    | uns longword        | read   | reference |
| value_mask     | mask_longword | uns longword        | read   | reference |
| control_values | record        | x\$keyboard_control | read   | reference |

---

## CHANGE KEYBOARD MAPPING

### OpenVMS Format

X\$CHANGE\_KEYBOARD\_MAPPING  
(display, first\_keycode, keysyms\_per\_keycode, keysyms\_ids, num\_keycodes)

### Argument Information

| Argument            | Usage      | Data Type    | Access | Mechanism |
|---------------------|------------|--------------|--------|-----------|
| display             | identifier | uns longword | read   | reference |
| first_keycode       | longword   | uns longword | read   | reference |
| keysyms_per_keycode | longword   | longword     | read   | reference |
| keysyms_ids         | array      | uns longword | read   | reference |
| num_keycodes        | longword   | longword     | read   | reference |

### Arguments

#### keysyms\_ids

A pointer to a list containing the specified key symbols for the key codes. The total number of key symbols specified must be a multiple of **keysyms\_per\_keycode**. The list is an array where each element contains a key symbol.

## Xlib Routines

### CHANGE POINTER CONTROL

---

## CHANGE POINTER CONTROL

### OpenVMS Format

X\$CHANGE\_POINTER\_CONTROL

(display, do\_accel, do\_threshold, accel\_numerator, accel\_denominator, threshold)

### Argument Information

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| display           | identifier | uns longword | read   | reference |
| do_accel          | Boolean    | longword     | read   | reference |
| do_threshold      | Boolean    | longword     | read   | reference |
| accel_numerator   | longword   | longword     | read   | reference |
| accel_denominator | longword   | longword     | read   | reference |
| threshold         | longword   | longword     | read   | reference |

---

## CHANGE PROPERTY

### OpenVMS Format

X\$CHANGE\_PROPERTY

(display, window\_id, property\_id, type\_id, format, change\_mode, prop\_data, num\_elements)

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| display      | identifier | uns longword | read   | reference |
| window_id    | identifier | uns longword | read   | reference |
| property_id  | identifier | uns longword | read   | reference |
| type_id      | identifier | uns longword | read   | reference |
| format       | longword   | longword     | read   | reference |
| change_mode  | longword   | uns longword | read   | reference |
| prop_data    | array      | byte         | read   | reference |
| num_elements | longword   | longword     | read   | reference |

## Arguments

### **change\_mode**

The type of property change to be completed by the routine. The predefined values for **change\_mode** are as follows:

XSC\_PROP\_MODE\_REPLACE  
XSC\_PROP\_MODE\_PREPEND  
XSC\_PROP\_MODE\_APPEND  
None

The value **None** assumes that the values for the **type\_id** and **format** specified match the values for the specified property, and that the new value is zero. Other values specified in this argument are not valid.

## CHANGE SAVE SET

### OpenVMS Format

X\$CHANGE\_SAVE\_SET  
(display, window\_id, change\_mode)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| window_id   | identifier | uns longword | read   | reference |
| change_mode | longword   | longword     | read   | reference |

## Arguments

### **change\_mode**

The predefined values for **change\_mode** are as follows:

| Value               | Description                                              |
|---------------------|----------------------------------------------------------|
| XSC_SET_MODE_INSERT | Adds the specified windows to the client's saveset.      |
| XSC_SET_MODE_DELETE | Removes the specified windows from the client's saveset. |

## Xlib Routines

### CHANGE WINDOW ATTRIBUTES

---

## CHANGE WINDOW ATTRIBUTES

### OpenVMS Format

X\$CHANGE\_WINDOW\_ATTRIBUTES  
(display, window\_id, attributes\_mask, attributes)

### Argument Information

| Argument        | Usage        | Data Type                 | Access | Mechanism |
|-----------------|--------------|---------------------------|--------|-----------|
| display         | identifier   | uns longword              | read   | reference |
| window_id       | identifier   | uns longword              | read   | reference |
| attributes_mask | uns longword | uns longword              | read   | reference |
| attributes      | record       | x\$set_win_<br>attributes | read   | reference |

---

## CHECK IF EVENT

### OpenVMS Format

present\_return = X\$CHECK\_IF\_EVENT  
(display, event\_return, predicate, arg)

### Argument Information

| Argument       | Usage      | Data Type       | Access | Mechanism |
|----------------|------------|-----------------|--------|-----------|
| present_return | Boolean    | longword        | write  | value     |
| display        | identifier | uns longword    | read   | reference |
| event_return   | record     | x\$event        | write  | reference |
| predicate      | procedure  | proc entry mask | read   | reference |
| arg            | longword   | uns longword    | read   | value     |

---

## CHECK MASK EVENT

### OpenVMS Format

present\_return = X\$CHECK\_MASK\_EVENT  
(display, event\_mask, event\_return)

**Argument Information**

| Argument       | Usage         | Data Type    | Access | Mechanism |
|----------------|---------------|--------------|--------|-----------|
| present_return | Boolean       | longword     | write  | value     |
| display        | identifier    | uns longword | read   | reference |
| event_mask     | mask_longword | longword     | read   | reference |
| event_return   | record        | x\$event     | write  | reference |

**CHECK TYPED EVENT**

**OpenVMS Format**

```
present_return = X$CHECK_TYPED_EVENT
    (display, event_type, event_return)
```

**Argument Information**

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| present_return | Boolean    | longword     | write  | value     |
| display        | identifier | uns longword | read   | reference |
| event_type     | longword   | longword     | read   | reference |
| event_return   | record     | x\$event     | write  | reference |

**CHECK TYPED WINDOW EVENT**

**OpenVMS Format**

```
present_return = X$CHECK_TYPED_WINDOW_EVENT
    (display, window_id, event_type, event_return)
```

**Argument Information**

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| present_return | Boolean    | longword     | write  | value     |
| display        | identifier | uns longword | read   | reference |
| window_id      | identifier | uns longword | read   | reference |
| event_type     | longword   | longword     | read   | reference |
| event_return   | record     | x\$event     | write  | reference |

## Xlib Routines

### CHECK WINDOW EVENT

---

## CHECK WINDOW EVENT

### OpenVMS Format

```
present_return = X$CHECK_WINDOW_EVENT
    (display, window_id, event_mask, event_return)
```

### Argument Information

| Argument       | Usage         | Data Type    | Access | Mechanism |
|----------------|---------------|--------------|--------|-----------|
| present_return | Boolean       | longword     | write  | value     |
| display        | identifier    | uns longword | read   | reference |
| window_id      | identifier    | uns longword | read   | reference |
| event_mask     | mask_longword | uns longword | read   | reference |
| event_return   | record        | x\$event     | write  | reference |

---

## CIRCULATE SUBWINDOWS

### OpenVMS Format

```
X$CIRCULATE_SUBWINDOWS
    (display, window_id, direction)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |
| direction | longword   | longword     | read   | reference |

### Arguments

#### direction

The direction in which you want to circulate the window. Valid values are listed as follows:

```
X$C_RAISE_LOWEST
X$C_LOWER_HIGHEST
```

## CIRCULATE SUBWINDOWS DOWN

### OpenVMS Format

X\$CIRCULATE\_SUBWINDOWS\_DOWN  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

## CIRCULATE SUBWINDOWS UP

### OpenVMS Format

X\$CIRCULATE\_SUBWINDOWS\_UP  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

## CLEAR AREA

### OpenVMS Format

X\$CLEAR\_AREA  
(display, window\_id, x\_coord, y\_coord, width, height, exposures)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |
| x_coord   | longword   | longword     | read   | reference |

## Xlib Routines

### CLEAR AREA

| Argument  | Usage    | Data Type    | Access | Mechanism |
|-----------|----------|--------------|--------|-----------|
| y_coord   | longword | longword     | read   | reference |
| width     | longword | uns longword | read   | reference |
| height    | longword | uns longword | read   | reference |
| exposures | Boolean  | uns longword | read   | reference |

---

## CLEAR WINDOW

### OpenVMS Format

X\$CLEAR\_WINDOW  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## CLIP BOX

### OpenVMS Format

X\$CLIP\_BOX  
(region\_id, rectangle\_struc\_return)

### Argument Information

| Argument               | usage      | Data Type    | Access | Mechanism |
|------------------------|------------|--------------|--------|-----------|
| region_id              | identifier | uns longword | read   | reference |
| rectangle_struc_return | record     | x\$rectangle | write  | reference |



## CLOSE DISPLAY

### OpenVMS Format

X\$CLOSE\_DISPLAY  
(display)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## CONFIGURE WINDOW

### OpenVMS Format

X\$CONFIGURE\_WINDOW  
(display, window\_id, change\_mask, values)

### Argument Information

| Argument    | Usage        | Data Type         | Access | Mechanism |
|-------------|--------------|-------------------|--------|-----------|
| display     | identifier   | uns longword      | read   | reference |
| window_id   | identifier   | uns longword      | read   | reference |
| change_mask | uns longword | uns longword      | read   | reference |
| values      | record       | x\$window_changes | read   | reference |

### Arguments

#### change\_mask

A bit mask that specifies the values that are to be set using the information in the data structure to which the **values** argument points.

Table 2–2 lists each bit for the **change\_mask** argument, its predefined value, and its description.

**Table 2–2 Change Mask Bits**

| Bit | Predefined Value | Description         |
|-----|------------------|---------------------|
| 0   | X\$C_CW_X        | Change x-coordinate |
| 1   | X\$C_CW_Y        | Change y-coordinate |
| 2   | X\$C_CW_WIDTH    | Change width        |

(continued on next page)

## Xlib Routines

### CONFIGURE WINDOW

Table 2-2 (Cont.) Change Mask Bits

| Bit | Predefined Value    | Description         |
|-----|---------------------|---------------------|
| 3   | XSC_CW_HEIGHT       | Change height       |
| 4   | XSC_CW_BORDER_WIDTH | Change border width |
| 5   | XSC_CW_SIBLING      | Change sibling      |
| 6   | XSC_CW_STACK_MODE   | Change stack mode   |
| 7   | NONE                | Reserved            |
| 8   | NONE                | Reserved            |

---

## CONNECTION NUMBER

### OpenVMS Format

```
int = X$CONNECTION_NUMBER  
(display)
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| int      | longword   | longword     | write  | value     |
| display  | identifier | uns longword | read   | reference |

---

#### Note

CONNECTION NUMBER returns an integer that identifies the connection. This routine is defined to be operating system specific by MIT, and there is no direct analogue to the UNIX file descriptor in OpenVMS.

---

---

## CONVERT SELECTION

### OpenVMS Format

```
X$CONVERT_SELECTION  
(display, selection_id, target_id, property_id, requestor_id, time)
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| selection_id | identifier | uns longword | read   | reference |
| target_id    | identifier | uns longword | read   | reference |
| property_id  | identifier | uns longword | read   | reference |
| requestor_id | identifier | uns longword | read   | reference |
| time         | longword   | uns longword | read   | reference |

---

## COPY AREA

### OpenVMS Format

X\$COPY\_AREA

(display, src\_drawable\_id, dst\_drawable\_id, gc\_id, src\_x\_coord, src\_y\_coord, width, height, dst\_x\_coord, dst\_y\_coord)

### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| display         | identifier | uns longword | read   | reference |
| src_drawable_id | identifier | uns longword | read   | reference |
| dst_drawable_id | identifier | uns longword | read   | reference |
| gc_id           | identifier | uns longword | read   | reference |
| src_x_coord     | longword   | longword     | read   | reference |
| src_y_coord     | longword   | longword     | read   | reference |
| width           | longword   | uns longword | read   | reference |
| height          | longword   | uns longword | read   | reference |
| dst_x_coord     | longword   | longword     | read   | reference |
| dst_y_coord     | longword   | longword     | read   | reference |

---

## COPY COLORMAP AND FREE

### OpenVMS Format

colormap\_id\_return = X\$COPY\_COLORMAP\_AND\_FREE

(display, colormap\_id)

## Xlib Routines

### COPY COLORMAP AND FREE

#### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| colormap_id_return | identifier | uns longword | write  | value     |
| display            | identifier | uns longword | read   | reference |
| colormap_id        | identifier | uns longword | read   | reference |

---

## COPY GC

### OpenVMS Format

X\$COPY\_GC  
(display, src\_gc\_id, gc\_mask, dst\_gc\_id)

#### Argument Information

| Argument  | Usage         | Data Type    | Access | Mechanism |
|-----------|---------------|--------------|--------|-----------|
| display   | identifier    | uns longword | read   | reference |
| src_gc_id | identifier    | uns longword | read   | reference |
| gc_mask   | mask_longword | uns longword | read   | reference |
| dst_gc_id | identifier    | uns longword | read   | reference |

---

## COPY PLANE

### OpenVMS Format

X\$COPY\_PLANE  
(display, src\_drawable\_id, dst\_drawable\_id, gc\_id, src\_x\_coord, src\_y\_coord, width, height, dst\_x\_coord, dst\_y\_coord, plane)

#### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| display         | identifier | uns longword | read   | reference |
| src_drawable_id | identifier | uns longword | read   | reference |
| dst_drawable_id | identifier | uns longword | read   | reference |
| gc_id           | identifier | uns longword | read   | reference |
| src_x_coord     | longword   | longword     | read   | reference |
| src_y_coord     | longword   | longword     | read   | reference |

| Argument    | Usage         | Data Type    | Access | Mechanism |
|-------------|---------------|--------------|--------|-----------|
| width       | longword      | uns longword | read   | reference |
| height      | longword      | uns longword | read   | reference |
| dst_x_coord | longword      | longword     | read   | reference |
| dst_y_coord | longword      | longword     | read   | reference |
| plane       | mask_longword | uns longword | read   | reference |

---

## CREATE BITMAP FROM DATA

### OpenVMS Format

```

pixmap_id_return = X$CREATE_BITMAP_FROM_DATA
    (display, drawable_id, data, width, height)

```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| pixmap_id_return | identifier | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| drawable_id      | identifier | uns longword | read   | reference |
| data             | array      | byte         | read   | reference |
| width            | longword   | uns longword | read   | reference |
| height           | longword   | uns longword | read   | reference |

---

## CREATE COLORMAP

### OpenVMS Format

```

colormap_id_return = X$CREATE_COLORMAP
    (display, window_id, visual_struct, alloc)

```

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| colormap_id_return | identifier | uns longword | write  | value     |
| display            | identifier | uns longword | read   | reference |
| window_id          | identifier | uns longword | read   | reference |
| visual_struct      | record     | x\$visual    | read   | reference |
| alloc              | longword   | longword     | read   | reference |

## Xlib Routines

### CREATE COLORMAP

#### Arguments

##### **alloc**

The allocation mode of color map entries. The values for **alloc** are as follows:

XSC\_ALLOC\_NONE

XSC\_ALLOC\_ALL

Other values specified in this argument are not valid.

If static gray, static color, or true color is specified in **visual\_struct**, no entries can be allocated from the color map.

---

## CREATE FONT CURSOR

#### OpenVMS Format

```
cursor_id_return = X$CREATE_FONT_CURSOR
```

```
(display, cursor_shape)
```

#### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| cursor_id_return | identifier | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| cursor_shape     | longword   | uns longword | read   | reference |

---

## CREATE GC

#### OpenVMS Format

```
gc_id_return = X$CREATE_GC
```

```
(display, drawable_id, gc_mask, values_struct)
```

#### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| gc_id_return  | identifier    | uns longword | write  | value     |
| display       | identifier    | uns longword | read   | reference |
| drawable_id   | identifier    | uns longword | read   | reference |
| gc_mask       | mask_longword | uns longword | read   | reference |
| values_struct | record        | x\$gc_values | read   | reference |

## CREATE GLYPH CURSOR

### OpenVMS Format

```
cursor_id_return = X$CREATE_GLYPH_CURSOR
    (display, src_font_id, mask_font_id, src_char, mask_char, foreground_color,
     background_color)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| cursor_id_return | identifier | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| src_font_id      | identifier | uns longword | read   | reference |
| mask_font_id     | identifier | uns longword | read   | reference |
| src_char         | longword   | uns longword | read   | reference |
| mask_char        | longword   | uns longword | read   | reference |
| foreground_color | record     | x\$color     | read   | reference |
| background_color | record     | x\$color     | read   | reference |

## CREATE IMAGE

### OpenVMS Format

```
status_return = X$CREATE_IMAGE
    (display, visual_struct, depth, image_format, offset, data, width, height, bitmap_pad,
     bytes_per_line, ximage_return)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | longword   | longword     | write  | value     |
| display       | identifier | uns longword | read   | reference |
| visual_struct | record     | x\$visual    | read   | reference |
| depth         | longword   | longword     | read   | reference |
| image_format  | longword   | longword     | read   | reference |
| offset        | longword   | longword     | read   | reference |
| data          | array      | byte         | read   | reference |
| width         | longword   | uns longword | read   | reference |
| height        | longword   | uns longword | read   | reference |

## Xlib Routines

### CREATE IMAGE

| Argument       | Usage    | Data Type | Access | Mechanism |
|----------------|----------|-----------|--------|-----------|
| bitmap_pad     | longword | longword  | read   | reference |
| bytes_per_line | longword | longword  | read   | reference |
| ximage_return  | record   | x\$image  | write  | reference |

### Returns

#### status\_return

Specifies whether the return completed successfully. CREATE IMAGE returns one of the following values:

| Value       | Description                     |
|-------------|---------------------------------|
| SS\$_NORMAL | Routine completed successfully. |
| 0           | Error received for failure.     |

### Arguments

#### image\_format

The format of the image that will be created and referenced by the image data structure. The predefined values for **image\_format** are as follows:

XSC\_XY\_BITMAP  
XSC\_XY\_PIXMAP  
XSC\_Z\_PIXMAP

Other values specified in this argument are not valid.

---

## CREATE PIXMAP

### OpenVMS Format

```
pixmap_id_return = X$CREATE_PIXMAP  
(display, drawable_id, width, height, depth)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| pixmap_id_return | identifier | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| drawable_id      | identifier | uns longword | read   | reference |
| width            | longword   | uns longword | read   | reference |
| height           | longword   | uns longword | read   | reference |
| depth            | longword   | longword     | read   | reference |



---

## CREATE PIXMAP FROM BITMAP DATA

### OpenVMS Format

```
 pixmap_id_return = X$CREATE_PIXMAP_FROM_BITMAP_DATA  
 (display, drawable_id, data, width, height, foreground, background, depth)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| pixmap_id_return | identifier | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| drawable_id      | identifier | uns longword | read   | reference |
| data             | array      | byte         | read   | reference |
| width            | longword   | uns longword | read   | reference |
| height           | longword   | uns longword | read   | reference |
| foreground       | longword   | uns longword | read   | reference |
| background       | longword   | uns longword | read   | reference |
| depth            | longword   | longword     | read   | reference |

---

## CREATE REGION

### OpenVMS Format

```
 region_id_return = X$CREATE_REGION
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| region_id_return | identifier | uns longword | write  | reference |

---

## CREATE SIMPLE WINDOW

### OpenVMS Format

```
 window_id_return = X$CREATE_SIMPLE_WINDOW  
 (display, parent_id, x_coord, y_coord, width, height, border_width, border_id,  
 background_id)
```

## Xlib Routines

### CREATE SIMPLE WINDOW

#### Argument Information

| Argument         | Usage        | Data Type    | Access | Mechanism |
|------------------|--------------|--------------|--------|-----------|
| window_id_return | identifier   | uns longword | write  | value     |
| display          | identifier   | uns longword | read   | reference |
| parent_id        | identifier   | uns longword | read   | reference |
| x_coord          | longword     | longword     | read   | reference |
| y_coord          | longword     | longword     | read   | reference |
| width            | uns longword | uns longword | read   | reference |
| height           | uns longword | uns longword | read   | reference |
| border_width     | uns longword | uns longword | read   | reference |
| border_id        | identifier   | uns longword | read   | reference |
| background_id    | identifier   | uns longword | read   | reference |

## CREATE WINDOW

#### OpenVMS Format

window\_id\_return = X\$CREATE\_WINDOW

(display, parent\_id, x\_coord, y\_coord, width, height, border\_width, depth, class,  
visual\_struc, attributes\_mask, attributes)

#### Argument Information

| Argument         | Usage        | Data Type             | Access | Mechanism |
|------------------|--------------|-----------------------|--------|-----------|
| window_id_return | identifier   | uns longword          | write  | reference |
| display          | identifier   | uns longword          | read   | reference |
| parent_id        | identifier   | uns longword          | read   | reference |
| x_coord          | longword     | longword              | read   | reference |
| y_coord          | longword     | longword              | read   | reference |
| width            | uns longword | uns longword          | read   | reference |
| height           | uns longword | uns longword          | read   | reference |
| border_width     | uns longword | uns longword          | read   | reference |
| depth            | longword     | longword              | read   | reference |
| class            | longword     | longword              | read   | reference |
| visual_struc     | record       | x\$visual             | read   | reference |
| attributes_mask  | uns longword | uns longword          | read   | reference |
| attributes       | record       | x\$set_win_attributes | read   | reference |

---

## DEFAULT COLORMAP

### OpenVMS Format

```
colormap_id_return = X$DEFAULT_COLORMAP
    (display, screen_number)
```

### Argument Information

| Argument           | Usage        | Data Type    | Access | Mechanism |
|--------------------|--------------|--------------|--------|-----------|
| colormap_id_return | identifier   | longword     | write  | value     |
| display            | identifier   | uns longword | read   | reference |
| screen_number      | uns longword | uns longword | read   | reference |

---

## DEFAULT COLORMAP OF SCREEN

### OpenVMS Format

```
colormap_id_return = X$DEFAULT_COLORMAP_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| colormap_id_return | identifier | uns longword | write  | value     |
| screen_id          | identifier | uns longword | read   | reference |

---

## DEFAULT DEPTH

### OpenVMS Format

```
depth_return = X$DEFAULT_DEPTH
    (display, screen_number)
```

### Argument Information

| Argument     | Usage        | Data Type    | Access | Mechanism |
|--------------|--------------|--------------|--------|-----------|
| depth_return | uns longword | uns longword | write  | value     |
| display      | identifier   | uns longword | read   | reference |

## Xlib Routines

### DEFAULT DEPTH

---

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| screen_number | uns longword | uns longword | read   | reference |

---

---

## DEFAULT DEPTH OF SCREEN

### OpenVMS Format

```
depth_return = X$DEFAULT_DEPTH_OF_SCREEN  
    (screen_id)
```

### Argument Information

---

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| depth_return | longword   | longword     | write  | value     |
| screen_id    | identifier | uns longword | read   | reference |

---

---

## DEFAULT GC

### OpenVMS Format

```
context_id_return = X$DEFAULT_GC  
    (display, screen_number)
```

### Argument Information

---

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| context_id_return | identifier | uns longword | write  | value     |
| display           | identifier | uns longword | read   | reference |
| screen_number     | identifier | uns longword | read   | reference |

---

---

## DEFAULT GC OF SCREEN

### OpenVMS Format

```
gc_return = X$DEFAULT_GC_OF_SCREEN  
    (screen_id)
```

**Argument Information**

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| gc_return | identifier | uns longword | write  | value     |
| screen_id | identifier | uns longword | read   | reference |

**DEFAULT ROOT WINDOW**

**OpenVMS Format**

```

window_id_return = X$DEFAULT_ROOT_WINDOW
    (display)

```

**Argument Information**

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| window_id_return | identifier | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |

**DEFAULT SCREEN**

**OpenVMS Format**

```

screen_number_return = X$DEFAULT_SCREEN
    (display)

```

**Argument Information**

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| screen_number_return | identifier | uns longword | write  | value     |
| display              | identifier | uns longword | read   | reference |

## Xlib Routines

### DEFAULT SCREEN OF DISPLAY

---

## DEFAULT SCREEN OF DISPLAY

### OpenVMS Format

```
screen_id_return = X$DEFAULT_SCREEN_OF_DISPLAY  
(display)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| screen_id_return | record     | x\$screen    | write  | value     |
| display          | identifier | uns longword | read   | reference |

---

## DEFAULT VISUAL

### OpenVMS Format

```
status_return = X$DEFAULT_VISUAL  
(display, screen_number, visual_return)
```

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| status_return | cond_value   | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |
| visual_return | record       | x\$visual    | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. DEFAULT VISUAL returns one of the following values:

| Value        | Description                     |
|--------------|---------------------------------|
| SS\$_NORMAL  | Routine completed successfully. |
| X\$_NOTFOUND | The default visual is null.     |

### Arguments

#### visual\_return

The returned visual structure.

---

## DEFAULT VISUAL OF SCREEN

### OpenVMS Format

```
X$DEFAULT_VISUAL_OF_SCREEN  
(screen_id, visual_return)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| screen_id     | identifier | uns longword | read   | reference |
| visual_return | record     | x\$visual    | write  | reference |

### Returns

**visual\_return**  
The returned visual structure.

---

## DEFINE CURSOR

### OpenVMS Format

```
X$DEFINE_CURSOR  
(display, window_id, cursor_id)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |
| cursor_id | identifier | uns longword | read   | reference |

---

## DELETE CONTEXT

### OpenVMS Format

```
status_return = X$DELETE_CONTEXT  
(display, window_id, context_id)
```

## Xlib Routines

### DELETE CONTEXT

#### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | longword   | longword     | write  | value     |
| display       | identifier | uns longword | read   | reference |
| window_id     | identifier | uns longword | read   | reference |
| context_id    | identifier | uns longword | read   | reference |

#### Returns

##### status\_return

Specifies whether the return completed successfully. The routine returns zero for success and a non-zero value for failure.

---

## DELETE MODIFIERMAP ENTRY

#### OpenVMS Format

X\$DELETE\_MODIFIERMAP\_ENTRY

(modifier\_keys, keycode\_entry, modifier, modifier\_keys\_return )

#### Argument Information

| Argument             | Usage      | Data Type          | Access | Mechanism |
|----------------------|------------|--------------------|--------|-----------|
| modifier_keys        | record     | x\$modifier_keymap | read   | reference |
| keycode_entry        | identifier | uns longword       | read   | reference |
| modifier             | longword   | uns longword       | read   | reference |
| modifier_keys_return | record     | x\$modifier_keymap | write  | reference |

#### Arguments

##### modifier

The modifier for which you delete a key symbol. There are eight modifiers in the order (starting from zero) shift, lock, control, mod1, mod2, mod3, mod4, and mod5. You can pass the integer value or one of the following constants:

X\$C\_SHIFT\_MAP\_INDEX  
X\$C\_LOCK\_MAP\_INDEX  
X\$C\_CONTROL\_MAP\_INDEX  
X\$C\_MOD1\_MAP\_INDEX  
X\$C\_MOD2\_MAP\_INDEX  
X\$C\_MOD3\_MAP\_INDEX  
X\$C\_MOD4\_MAP\_INDEX  
X\$C\_MOD5\_MAP\_INDEX



**modifier\_keys\_return**

A pointer to a modifier keys structure. DELETE MODIFIERMAP ENTRY returns the revised modifier key map structure to this client-supplied structure.

---

## DELETE PROPERTY

### OpenVMS Format

X\$DELETE\_PROPERTY

(display, window\_id, property\_id)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| window_id   | identifier | uns longword | read   | reference |
| property_id | identifier | uns longword | read   | reference |

---

## DESTROY IMAGE

### OpenVMS Format

X\$DESTROY\_IMAGE

(ximage)

### Argument Information

| Argument | Usage  | Data Type | Access | Mechanism |
|----------|--------|-----------|--------|-----------|
| ximage   | record | x\$image  | read   | reference |

---

## DESTROY REGION

### OpenVMS Format

X\$DESTROY\_REGION

(region\_id)

## Xlib Routines

### DESTROY REGION

#### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| region_id | identifier | uns longword | read   | reference |

---

## DESTROY SUBWINDOWS

#### OpenVMS Format

X\$DESTROY\_SUBWINDOWS  
(display, window\_id)

#### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## DESTROY WINDOW

#### OpenVMS Format

X\$DESTROY\_WINDOW  
(display, window\_id)

#### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## DISABLE ACCESS CONTROL

#### OpenVMS Format

X\$DISABLE\_ACCESS\_CONTROL  
(display)

**Argument Information**

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## DISPLAY CELLS

**OpenVMS Format**

cells\_return = X\$DISPLAY\_CELLS  
(display, screen\_number)

**Argument Information**

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| cells_return  | longword     | longword     | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |

## DISPLAY HEIGHT

**OpenVMS Format**

height\_return = X\$DISPLAY\_HEIGHT  
(display, screen\_number)

**Argument Information**

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| height_return | uns longword | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |

## Xlib Routines

### DISPLAY HEIGHT MM

---

## DISPLAY HEIGHT MM

### OpenVMS Format

height\_return = X\$DISPLAY\_HEIGHT\_MM  
(display, screen\_number)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| height_return | uns longword | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |

---

## DISPLAY KEYCODES

### OpenVMS Format

X\$DISPLAY\_KEYCODES  
(display, [min\_keycodes\_return], [max\_keycodes\_return])

### Argument Information

| Argument            | Usage      | Data Type    | Access | Mechanism |
|---------------------|------------|--------------|--------|-----------|
| display             | identifier | uns longword | read   | reference |
| min_keycodes_return | longword   | longword     | write  | reference |
| max_keycodes_return | longword   | longword     | write  | reference |

---

## DISPLAY MOTION BUFFER SIZE

### OpenVMS Format

size\_return = X\$DISPLAY\_MOTION\_BUFFER\_SIZE  
(display)

**Argument Information**

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| size_return | longword   | uns longword | write  | value     |
| display     | identifier | uns longword | read   | reference |

---

**DISPLAY NAME**

**OpenVMS Format**

```
status_return = X$DISPLAY_NAME
      (disp_string, disp_name_return, [disp_len_return])
```

**Argument Information**

| Argument         | Usage       | Data Type        | Access | Mechanism  |
|------------------|-------------|------------------|--------|------------|
| status_return    | cond_value  | uns longword     | write  | value      |
| disp_string      | char_string | character string | read   | descriptor |
| disp_name_return | char_string | character string | write  | descriptor |
| disp_len_return  | word        | uns word         | write  | reference  |

**Returns**

**status\_return**

Specifies whether or not the routine completed successfully. The argument **status\_return** can be set to any of the OpenVMS condition codes returned by LIB\$SCOPY\_R\_DX.

**disp\_string**

The address of a character string descriptor that points to the display name. If the descriptor address is zero or points to a character string of zero length, the DISPLAY NAME function translates the logical name DECW\$DISPLAY to find the display name.

**disp\_name\_return**

The address of a character string descriptor that points to the string. DISPLAY NAME returns the display name to this argument.

**disp\_len\_return**

DISPLAY NAME returns the length of the display name string to this optional argument.

## Xlib Routines

### DISPLAY OF SCREEN

---

## DISPLAY OF SCREEN

### OpenVMS Format

X\$DISPLAY\_OF\_SCREEN  
(screen\_id, display\_return)

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| screen_id      | identifier | uns longword | read   | reference |
| display_return | record     | x\$display   | write  | value     |

### Arguments

**display\_return**  
The display of the specified screen.

---

## DISPLAY PLANES

### OpenVMS Format

planes\_return = X\$DISPLAY\_PLANES  
(display, screen\_number)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| planes_return | longword     | longword     | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |

---

## DISPLAY STRING

### OpenVMS Format

status\_return = X\$DISPLAY\_STRING  
(display, name\_return, [len\_return])

**Argument Information**

| Argument      | Usage         | Data Type        | Access | Mechanism  |
|---------------|---------------|------------------|--------|------------|
| status_return | cond_value    | uns longword     | write  | value      |
| display       | identifier    | uns longword     | read   | reference  |
| name_return   | char_string   | character string | write  | descriptor |
| len_return    | unsigned_word | uns word         | write  | reference  |

**Arguments**

**name\_return**

The name of the string passed to OPEN DISPLAY when the current display was opened.

The **name\_return** argument is the address of a character string descriptor that points to the string.

**len\_return**

The length of the returned string. This argument is optional.

**Returns**

**status\_return**

Specifies whether or not the routine completed successfully. The argument **status\_return** can be set to any of the OpenVMS condition codes returned by LIB\$SCOPY\_R\_DX.

**DISPLAY WIDTH**

**OpenVMS Format**

```
width_return = X$DISPLAY_WIDTH
    (display, screen_number)
```

**Argument Information**

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| width_return  | uns longword | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |

## Xlib Routines

### DISPLAY WIDTH MM

---

## DISPLAY WIDTH MM

### OpenVMS Format

```
width_return = X$DISPLAY_WIDTH_MM  
(display, screen_id)
```

### Argument Information

| Argument     | Usage        | Data Type    | Access | Mechanism |
|--------------|--------------|--------------|--------|-----------|
| width_return | uns longword | uns longword | write  | value     |
| display      | identifier   | uns longword | read   | reference |
| screen_id    | identifier   | uns longword | read   | reference |

---

## DOES BACKING STORE

### OpenVMS Format

```
support_value = X$DOES_BACKING_STORE  
(screen_id)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| support_value | longword   | longword     | write  | value     |
| screen_id     | identifier | uns longword | read   | reference |

### Returns

#### **support\_value**

Indicates whether the screen supports backing stores. This can be one of the following values:

```
X$C_ALWAYS  
X$C_WHEN_MAPPED  
X$C_NOT_USEFUL
```



---

## DOES SAVE UNDERS

### OpenVMS Format

```
Bool = X$DOES_SAVE_UNDERS
      (screen_id)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| Bool      | Boolean    | uns longword | write  | value     |
| screen_id | identifier | uns longword | read   | reference |

---

## DRAW ARC

### OpenVMS Format

```
X$DRAW_ARC
      (display, drawable_id, gc_id, x_coord, y_coord, width, height, angle1, angle2)
```

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |
| width       | longword   | uns longword | read   | reference |
| height      | longword   | uns longword | read   | reference |
| angle1      | longword   | longword     | read   | reference |
| angle2      | longword   | longword     | read   | reference |

---

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**, **width**, **height**, **angle1**, **angle2**.

---

---

## DRAW ARCS

### OpenVMS Format

X\$DRAW\_ARCS

(display, drawable\_id, gc\_id, arcs, num\_arcs)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| arcs        | array      | x\$arc       | read   | reference |
| num_arcs    | longword   | longword     | read   | reference |

---

## DRAW IMAGE STRING

### OpenVMS Format

X\$DRAW\_IMAGE\_STRING

(display, drawable\_id, gc\_id, x\_coord, y\_coord, string)

### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| display     | identifier  | uns longword     | read   | reference  |
| drawable_id | identifier  | uns longword     | read   | reference  |
| gc_id       | identifier  | uns longword     | read   | reference  |
| x_coord     | longword    | longword         | read   | reference  |
| y_coord     | longword    | longword         | read   | reference  |
| string      | char_string | character string | read   | descriptor |

---

#### Note

---

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

---

## ARGUMENTS

### **string**

The text string to be written to the screen. The **string** argument is the address of a descriptor that points to a string.

## DRAW IMAGE STRING 16

### OpenVMS Format

X\$DRAW\_IMAGE\_STRING\_16

(display, drawable\_id, gc\_id, x\_coord, y\_coord, string16, length)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |
| string16    | array      | word         | read   | reference |
| length      | word       | uns word     | read   | reference |

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

## DRAW LINE

### OpenVMS Format

X\$DRAW\_LINE

(display, drawable\_id, gc\_id, x1\_coord, y1\_coord, x2\_coord, y2\_coord)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## Xlib Routines

### DRAW LINE

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x1_coord    | longword   | longword     | read   | reference |
| y1_coord    | longword   | longword     | read   | reference |
| x2_coord    | longword   | longword     | read   | reference |
| y2_coord    | longword   | longword     | read   | reference |

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x1\_coord**, **y1\_coord**, **x2\_coord**, and **y2\_coord**.

---

## DRAW LINES

### OpenVMS Format

X\$DRAW\_LINES

(display, drawable\_id, gc\_id, points, num\_points, line\_mode)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| points      | array      | x\$point     | read   | reference |
| num_points  | longword   | longword     | read   | reference |
| line_mode   | longword   | longword     | read   | reference |

### Arguments

#### line\_mode

The coordinate mode of the points. The coordinates of the points can be relative to the drawable origin or to the previous point. The predefined values for **line\_mode** are as follows:

XSC\_COORD\_MODE\_ORIGIN  
XSC\_COORD\_MODE\_PREVIOUS

Other values specified in this argument are not valid.

---

## DRAW POINT

### OpenVMS Format

X\$DRAW\_POINT

(display, drawable\_id, gc\_id, x\_coord, y\_coord)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |

---

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

---



---

## DRAW POINTS

### OpenVMS Format

X\$DRAW\_POINTS

(display, drawable\_id, gc\_id, points, num\_points, point\_mode)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| points      | array      | x\$point     | read   | reference |
| num_points  | longword   | longword     | read   | reference |
| point_mode  | longword   | longword     | read   | reference |

## Xlib Routines DRAW POINTS

### Arguments

#### **point\_mode**

The coordinate mode of the points. The coordinates of the points can be relative to the drawable origin or relative to the previous point. The predefined values for **point\_mode** are as follows:

XSC\_COORD\_MODE\_ORIGIN  
XSC\_COORD\_MODE\_PREVIOUS

Other values specified in this argument are not valid.

---

## DRAW RECTANGLE

### OpenVMS Format

X\$DRAW\_RECTANGLE

(display, drawable\_id, gc\_id, x\_coord, y\_coord, width, height)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |
| width       | longword   | uns longword | read   | reference |
| height      | longword   | uns longword | read   | reference |

---

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**, **width**, **height**.

---

---

## DRAW RECTANGLES

### OpenVMS Format

X\$DRAW\_RECTANGLES

(display, drawable\_id, gc\_id, rectangles, num\_rectangles)

**Argument Information**

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| display        | identifier | uns longword | read   | reference |
| drawable_id    | identifier | uns longword | read   | reference |
| gc_id          | identifier | uns longword | read   | reference |
| rectangles     | array      | x\$rectangle | read   | reference |
| num_rectangles | longword   | longword     | read   | reference |

**DRAW SEGMENTS**

**OpenVMS Format**

X\$DRAW\_SEGMENTS

(display, drawable\_id, gc\_id, segments, num\_segments)

**Argument Information**

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| display      | identifier | uns longword | read   | reference |
| drawable_id  | identifier | uns longword | read   | reference |
| gc_id        | identifier | uns longword | read   | reference |
| segments     | array      | x\$segment   | read   | reference |
| num_segments | longword   | longword     | read   | reference |

**DRAW STRING**

**OpenVMS Format**

X\$DRAW\_STRING

(display, drawable\_id, gc\_id, x\_coord, y\_coord, string)

**Argument Information**

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |

## Xlib Routines

### DRAW STRING

| Argument | Usage       | Data Type        | Access | Mechanism  |
|----------|-------------|------------------|--------|------------|
| y_coord  | longword    | longword         | read   | reference  |
| string   | char_string | character string | read   | descriptor |

---

#### Note

---

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

---

---

## DRAW STRING 16

### OpenVMS Format

X\$DRAW\_STRING\_16

(display, drawable\_id, gc\_id, x\_coord, y\_coord, string16, length)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |
| string16    | array      | word         | read   | reference |
| length      | word       | word         | read   | reference |

---

#### Note

---

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

---

---

## DRAW TEXT

### OpenVMS Format

X\$DRAW\_TEXT

(display, drawable\_id, gc\_id, x\_coord, y\_coord, items, num\_items)



**Argument Information**

| Argument    | Usage      | Data Type       | Access | Mechanism |
|-------------|------------|-----------------|--------|-----------|
| display     | identifier | uns longword    | read   | reference |
| drawable_id | identifier | uns longword    | read   | reference |
| gc_id       | identifier | uns longword    | read   | reference |
| x_coord     | longword   | longword        | read   | reference |
| y_coord     | longword   | longword        | read   | reference |
| items       | array      | vector_longword | read   | reference |
| num_items   | longword   | longword        | read   | reference |

**Note**

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

---

**DRAW TEXT 16**

**OpenVMS Format**

X\$DRAW\_TEXT\_16

(display, drawable\_id, gc\_id, x\_coord, y\_coord, items16, num\_items)

**Argument Information**

| Argument    | Usage      | Data Type       | Access | Mechanism |
|-------------|------------|-----------------|--------|-----------|
| display     | identifier | uns longword    | read   | reference |
| drawable_id | identifier | uns longword    | read   | reference |
| gc_id       | identifier | uns longword    | read   | reference |
| x_coord     | longword   | longword        | read   | reference |
| y_coord     | longword   | longword        | read   | reference |
| items16     | array      | vector_longword | read   | reference |
| num_items   | longword   | longword        | read   | reference |

**Note**

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**.

## Xlib Routines

### EMPTY REGION

---

## EMPTY REGION

### OpenVMS Format

```
answer_return = X$EMPTY_REGION  
(region_id)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| answer_return | longword   | longword     | write  | value     |
| region_id     | identifier | uns longword | read   | reference |

---

## ENABLE ACCESS CONTROL

### OpenVMS Format

```
X$ENABLE_ACCESS_CONTROL  
(display)
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## EQUAL REGION

### OpenVMS Format

```
answer_return = X$EQUAL_REGION  
(region1_id, region2_id)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| answer_return | longword   | longword     | write  | value     |
| region1_id    | identifier | uns longword | read   | reference |
| region2_id    | identifier | uns longword | read   | reference |

## EVENT MASK OF SCREEN

### OpenVMS Format

```
mask_return = X$EVENT_MASK_OF_SCREEN
(screen_id)
```

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| mask_return | longword   | longword     | write  | value     |
| screen_id   | identifier | uns longword | read   | reference |

## EVENTS QUEUED

### OpenVMS Format

```
count = X$EVENTS_QUEUED
(display, mode)
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| count    | longword   | longword     | write  | value     |
| display  | identifier | uns longword | read   | reference |
| mode     | longword   | longword     | read   | reference |

### Arguments

#### mode

The mode by which you return the number of queued events. The predefined values for **mode** are as follows:

```
X$C_QUEUED_ALREADY
X$C_QUEUED_AFTER_FLUSH
X$C_QUEUED_AFTER_READING
```

## Xlib Routines

### FETCH BUFFER

---

## FETCH BUFFER

### OpenVMS Format

status\_return = X\$FETCH\_BUFFER  
(display, num\_bytes\_return, num\_buffer, buff\_addr\_return)

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| status_return    | cond_value | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| num_bytes_return | longword   | longword     | write  | reference |
| num_buffer       | longword   | longword     | read   | reference |
| buff_addr_return | longword   | longword     | write  | value     |

### Returns

**status\_return**  
Specifies whether the routine completed successfully. This routine returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| SS\$_NORMAL    | Routine completed successfully. |
| X\$_INVBUFFER  | Invalid buffer.                 |
| X\$_ERRORREPLY | Error received from server.     |
| X\$_INSFMEM    | Insufficient virtual memory.    |
| 0              | No data in the buffer.          |

### Arguments

**buff\_addr\_return**  
The address of the stream of bytes.

---

## FETCH BYTES

### OpenVMS Format

buff\_addr\_return = X\$FETCH\_BYTES  
(display, num\_bytes\_return)

**Argument Information**

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| buff_addr_return | longword   | longword     | write  | value     |
| display          | identifier | uns longword | read   | reference |
| num_bytes_return | longword   | longword     | write  | reference |

**Returns**

**buff\_addr\_return**  
The address of a stream of bytes stored in cut buffer zero.

**FETCH NAME**

**OpenVMS Format**

status\_return = X\$FETCH\_NAME  
(display, window\_id, window\_name\_return, [name\_len\_return])

**Argument Information**

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| status_return      | longword    | longword         | write  | value      |
| display            | identifier  | uns longword     | read   | reference  |
| window_id          | identifier  | uns longword     | read   | reference  |
| window_name_return | char_string | character string | write  | descriptor |
| name_len_return    | word        | uns word         | write  | reference  |

**Returns**

**status\_return**  
Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| 0        | Routine completed successfully.        |
| Non-zero | Routine did not complete successfully. |

**Arguments**

**name\_len\_return**  
The length of **window\_name\_return**. This argument is optional.

---

## FILL ARC

### OpenVMS Format

X\$FILL\_ARC

(display, drawable\_id, gc\_id, x\_coord, y\_coord, width, height, angle1, angle2)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |
| width       | longword   | uns longword | read   | reference |
| height      | longword   | uns longword | read   | reference |
| angle1      | longword   | longword     | read   | reference |
| angle2      | longword   | longword     | read   | reference |

---

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**, **width**, **height**, **angle1**, **angle2**.

---

---

## FILL ARCS

### OpenVMS Format

X\$FILL\_ARCS

(display, drawable\_id, gc\_id, arcs, num\_arcs)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| arcs        | array      | x\$arc       | read   | reference |
| num_arcs    | longword   | longword     | read   | reference |

---

## FILL POLYGON

### OpenVMS Format

X\$FILL\_POLYGON

(display, drawable\_id, gc\_id, points, num\_points, shape, mode)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| points      | array      | x\$point     | read   | reference |
| num_points  | longword   | longword     | read   | reference |
| shape       | longword   | longword     | read   | reference |
| mode        | longword   | longword     | read   | reference |

### Arguments

#### shape

The shape of the polygon. The predefined values for **shape** are as follows:

XSC\_POLYCOMPLEX  
XSC\_CONVEX  
XSC\_NONCONVEX

If the shape of the polygon is known and correctly specified with this argument, server performance can be improved. If the shape is specified incorrectly, the result of the operation will be undefined.

Other values specified in this argument are not valid.

#### mode

The coordinate mode of the points. The coordinates of the points can be relative to the drawable origin or relative to the previous point. The predefined values for **mode** are as follows:

XSC\_COORD\_MODE\_ORIGIN  
XSC\_COORD\_MODE\_PREVIOUS

Other values specified in this argument are not valid.

## Xlib Routines

### FILL RECTANGLE

---

## FILL RECTANGLE

### OpenVMS Format

X\$FILL\_RECTANGLE

(display, drawable\_id, gc\_id, x\_coord, y\_coord, width, height)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| x_coord     | longword   | longword     | read   | reference |
| y_coord     | longword   | longword     | read   | reference |
| width       | longword   | uns longword | read   | reference |
| height      | longword   | uns longword | read   | reference |

---

#### Note

The X Protocol defines that only the least significant 16 bits of the following longword arguments are transmitted to the server: **x\_coord**, **y\_coord**, **width**, **height**.

---

---

## FILL RECTANGLES

### OpenVMS Format

X\$FILL\_RECTANGLES

(display, drawable\_id, gc\_id, rectangles, num\_rectangles)

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| display        | identifier | uns longword | read   | reference |
| drawable_id    | identifier | uns longword | read   | reference |
| gc_id          | identifier | uns longword | read   | reference |
| rectangles     | array      | x\$rectangle | read   | reference |
| num_rectangles | longword   | longword     | read   | reference |



---

## **FILTER EVENT**

### **OpenVMS Format**

X\$FILTER\_EVENT  
(event, window)

### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| event           | record       | x\$event         | read          | reference        |
| window          | identifier   | uns longword     | read          | reference        |

---

## **FIND CONTEXT**

### **OpenVMS Format**

status\_return = X\$FIND\_CONTEXT  
(display, window\_id, context\_id, [window\_data\_return], [buff\_len], [buff\_ptr\_return], [len\_return])

### **Argument Information**

| <b>Argument</b>    | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|--------------------|--------------|------------------|---------------|------------------|
| status_return      | cond_value   | uns longword     | write         | value            |
| display            | identifier   | uns longword     | read          | reference        |
| window_id          | identifier   | uns longword     | read          | reference        |
| context_id         | identifier   | uns longword     | read          | reference        |
| window_data_return | longword     | uns longword     | read          | reference        |
| buff_len           | longword     | longword         | read          | reference        |
| buff_ptr_return    | array        | byte             | write         | reference        |
| len_return         | word         | uns word         | write         | reference        |

### **Returns**

**status\_return**  
Specifies whether the return completed successfully. The routine returns zero for success and a non-zero value for failure.

### **Arguments**

**buff\_len**  
The length of the supplied buffer. This argument is optional.

## Xlib Routines

### FIND CONTEXT

#### **buff\_ptr\_return**

The buffer into which data is written. This argument is optional.

#### **len\_return**

The length of the data written into the buffer. This argument is optional.

---

## FLUSH

### OpenVMS Format

X\$FLUSH

(display)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## FLUSH GC

### OpenVMS Format

X\$FLUSH\_GC

(display, gc)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc       | identifier | uns longword | read   | reference |

---

## FORCE SCREEN SAVER

### OpenVMS Format

X\$FORCE\_SCREEN\_SAVER

(display, saver\_mode)

**Argument Information**

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| saver_mode | longword   | longword     | read   | reference |

**Arguments**

**saver\_mode**

Specifies how the screen saver is activated. The predefined values for **saver\_mode** are as follows:

| Value                    | Description                                              |
|--------------------------|----------------------------------------------------------|
| X\$C_SCREEN_SAVER_ACTIVE | Activates the screen saver even if it has been disabled. |
| X\$C_SCREEN_SAVER_RESET  | Resets the screen saver to its initial state.            |

---

**FREE**

**OpenVMS Format**

X\$FREE  
(buff\_ptr)

**Argument Information**

| Argument | Usage | Data Type  | Access | Mechanism |
|----------|-------|------------|--------|-----------|
| buff_ptr | byte  | v uns byte | read   | reference |

---

**FREE COLORMAP**

**OpenVMS Format**

X\$FREE\_COLORMAP  
(display, colormap\_id)

## Xlib Routines FREE COLORMAP

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| colormap_id | identifier | uns longword | read   | reference |

---

## FREE COLORS

### OpenVMS Format

X\$FREE\_COLORS

(display, colormap\_id, pixels, num\_pixels, planes)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| colormap_id | identifier | uns longword | read   | reference |
| pixels      | array      | longword     | read   | reference |
| num_pixels  | longword   | longword     | read   | reference |
| planes      | longword   | uns longword | read   | reference |

---

## FREE CURSOR

### OpenVMS Format

X\$FREE\_CURSOR

(display, cursor\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| cursor_id | identifier | uns longword | read   | reference |

---

## **FREE FONT**

### **OpenVMS Format**

X\$FREE\_FONT  
(display, font\_struct)

#### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| display         | identifier   | uns longword     | read          | reference        |
| font_struct     | record       | x\$font_struct   | read          | reference        |

---

## **FREE GC**

### **OpenVMS Format**

X\$FREE\_GC  
(display, gc\_id)

#### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| display         | identifier   | uns longword     | read          | reference        |
| gc_id           | identifier   | uns longword     | read          | reference        |

---

## **FREE MODIFIERMAP**

### **OpenVMS Format**

X\$FREE\_MODIFIERMAP  
(modifier\_keys)

#### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b>   | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|--------------------|---------------|------------------|
| modifier_keys   | record       | x\$modifier_keymap | read          | reference        |

---

## FREE PIXMAP

### OpenVMS Format

X\$FREE\_PIXMAP  
(display, pixmap\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| pixmap_id | identifier | uns longword | read   | reference |

---

## FREE STRING LIST

### OpenVMS Format

X\$FREE\_STRING\_LIST  
(string\_list)

### Argument Information

| Argument    | Usage   | Data Type | Access | Mechanism |
|-------------|---------|-----------|--------|-----------|
| string_list | address | byte      | read   | reference |

---

## GCONTEXT FROM GC

### OpenVMS Format

gc\_resource\_id\_return = X\$GCONTEXT\_FROM\_GC  
(gc\_id)

### Argument Information

| Argument              | Usage      | Data Type    | Access | Mechanism |
|-----------------------|------------|--------------|--------|-----------|
| gc_resource_id_return | identifier | uns longword | write  | value     |
| gc_id                 | identifier | uns longword | read   | reference |

## GEOMETRY

### OpenVMS Format

mask\_return = X\$GEOMETRY

(display, screen\_number, position, default\_position, border\_width, font\_width, font\_height, xadd, yadd, [x\_coord\_return], [y\_coord\_return], [width\_return], [height\_return])

### Argument Information

| Argument         | Usage         | Data Type        | Access | Mechanism  |
|------------------|---------------|------------------|--------|------------|
| mask_return      | mask_longword | uns longword     | write  | value      |
| display          | identifier    | uns longword     | read   | reference  |
| screen_number    | longword      | uns longword     | read   | reference  |
| position         | char_string   | character string | read   | descriptor |
| default_position | char_string   | character string | read   | descriptor |
| border_width     | longword      | uns longword     | read   | reference  |
| font_width       | longword      | longword         | read   | reference  |
| font_height      | longword      | longword         | read   | reference  |
| xadd             | longword      | longword         | read   | reference  |
| yadd             | longword      | longword         | read   | reference  |
| x_coord_return   | longword      | longword         | write  | reference  |
| y_coord_return   | longword      | longword         | write  | reference  |
| width_return     | longword      | uns longword     | write  | reference  |
| height_return    | longword      | uns longword     | write  | reference  |

### Returns

#### mask\_return

A bit mask that specifies which of four values (width, height, x offset, y offset) are actually found in the string, and whether the x and y values are negative. Each bit indicates whether the corresponding value was found in the parsed string. For each value found, the corresponding argument is updated; for each value not found, the argument is left unchanged.

Table 2-3 lists the predefined values and descriptions for **mask\_return**.

**Table 2-3 Parse Mask Bits**

| Bit | Predefined Value | Description                                 |
|-----|------------------|---------------------------------------------|
| 1   | X\$M_NO_VALUE    | Reserved.                                   |
| 2   | X\$M_X_VALUE     | The x-coordinate of the origin of a window. |
| 3   | X\$M_Y_VALUE     | The y-coordinate of the origin of a window. |

(continued on next page)

## Xlib Routines GEOMETRY

Table 2–3 (Cont.) Parse Mask Bits

| Bit | Predefined Value      | Description                                |
|-----|-----------------------|--------------------------------------------|
| 4   | X\$M_WIDTH_VALUE      | The width of the window in pixels.         |
| 5   | X\$M_HEIGHT_VALUE     | The height of the window in pixels.        |
| 6   | X\$M_ALL_VALUES       | Indicates if all values are present.       |
| 7   | X\$M_X_NEGATIVE_VALUE | Indicates if the x-coordinate is negative. |
| 8   | X\$M_Y_NEGATIVE_VALUE | Indicates if the y-coordinate is negative. |

### Arguments

#### **position**

The position string that you want to parse. The **position** argument is the address of a character string descriptor that points to the string.

#### **default\_position**

The default geometry specification string that you want to parse. The **default\_position** argument is the address of a character string descriptor that points to the string.

#### **x\_coord\_return**

The x-coordinate to which GEOMETRY returns the x-offset from the specified string and is relative to the origin of the drawable. This argument is optional.

#### **y\_coord\_return**

The y-coordinate to which GEOMETRY returns the y-offset from the specified string and is relative to the origin of the drawable. This argument is optional.

#### **width\_return**

Specifies the width value. This argument is optional.

#### **height\_return**

Specifies the height value. This argument is optional.

---

## GET ATOM NAME

### OpenVMS Format

```
status_return = X$GET_ATOM_NAME  
(display, atom_id, name_return, [name_len_return])
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| atom_id       | identifier | uns longword | read   | reference |



| Argument        | Usage       | Data Type        | Access | Mechanism  |
|-----------------|-------------|------------------|--------|------------|
| name_return     | char_string | character string | write  | descriptor |
| name_len_return | longword    | longword         | write  | reference  |

## Returns

### status\_return

Specifies whether the routine completed successfully. The argument **status\_return** can be set to any of the OpenVMS condition codes returned by LIB\$SCOPY\_R\_DX.

The routine returns one of the following condition values:

| Value         | Description                     |
|---------------|---------------------------------|
| SS\$NORMAL    | Routine completed successfully. |
| X\$ERRORREPLY | Error received from server.     |
| X\$INSFMEM    | Insufficient virtual memory.    |

## Arguments

### name\_return

The name of the atom associated with the identifier specified in **atom\_id**. The atom name is returned by the routine. The **name\_return** argument is the address of a character string descriptor that points to the string.

### name\_len\_return

The length of the atom name returned by the routine. This argument is optional.

---

## GET CHAR STRUCT

### OpenVMS Format

```
X$GET_CHAR_STRUCT
(font_info, char_code, char_struct)
```

### Argument Information

| Argument    | Usage    | Data Type      | Access | Mechanism |
|-------------|----------|----------------|--------|-----------|
| font_info   | record   | x\$font_struct | read   | reference |
| char_code   | longword | uns longword   | read   | reference |
| char_struct | record   | x\$char_struct | read   | reference |

## Xlib Routines

### GET CLASS HINT

---

## GET CLASS HINT

### OpenVMS Format

```
status_return = X$GET_CLASS_HINT  
(display, window_id, class_hints_return)
```

### Argument Information

| Argument           | Usage      | Data Type     | Access | Mechanism |
|--------------------|------------|---------------|--------|-----------|
| status_return      | cond_value | uns longword  | write  | value     |
| display            | identifier | uns longword  | read   | reference |
| window_id          | identifier | uns longword  | read   | reference |
| class_hints_return | record     | x\$class_hint | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully for the specified window. This argument returns one of the following values:

| Value         | Description                        |
|---------------|------------------------------------|
| SS\$_NORMAL   | Routine completed successfully.    |
| X\$_PROPUNDEF | The property has not been defined. |

---

## GET COMMAND

### OpenVMS Format

```
status_return = X$GET_COMMAND  
(display, window_id, argv, argc, [name_len])
```

### Argument Information

| Argument      | Usage        | Data Type        | Access | Mechanism |
|---------------|--------------|------------------|--------|-----------|
| status_return | cond_value   | uns longword     | write  | value     |
| display       | identifier   | uns longword     | read   | reference |
| window_id     | identifier   | uns longword     | read   | reference |
| argv          | char_string  | character string | read   | reference |
| argc          | uns longword | uns longword     | read   | reference |
| name_len      | uns longword | uns longword     | read   | reference |

## Returns

### **status\_return**

Specifies whether the routine completed successfully. The argument **status\_return** can be set to any of the OpenVMS condition codes returned by LIB\$SCOPY\_R\_DX. This argument returns one of the following values:

| Value      | Description                     |
|------------|---------------------------------|
| SS\$NORMAL | Routine completed successfully. |
| 0          | Property code does not exist.   |

## Arguments

### **name\_len**

Length of the returned command string. This argument is optional.

## GET DEFAULT

### OpenVMS Format

```
status_return = X$GET_DEFAULT
    (display, program_name, option_name, default_name_return, [default_len_return])
```

### Argument Information

| Argument            | Usage       | Data Type        | Access | Mechanism  |
|---------------------|-------------|------------------|--------|------------|
| status_return       | cond_value  | uns longword     | write  | value      |
| display             | identifier  | uns longword     | read   | reference  |
| program_name        | char_string | character string | read   | descriptor |
| option_name         | char_string | character string | read   | descriptor |
| default_name_return | char_string | character string | write  | descriptor |
| default_len_return  | word        | uns word         | write  | reference  |

## Returns

### **status\_return**

Specifies whether the routine completed successfully.

## Arguments

### **program\_name**

Name of the program that specifies the default property string for the user environment. The **program\_name** argument is the address of a character string descriptor that points to the string.

## Xlib Routines

### GET DEFAULT

#### **option\_name**

Name of the property option for determining the user environment defaults. The **option\_name** argument is the address of a character string descriptor that points to the string.

#### **default\_name\_return**

The address of a character string descriptor that points to the default property string.

#### **default\_len\_return**

The length of the default string minus any padding characters added to fill the string. This argument is optional.

---

## GET ERROR DATABASE TEXT

### OpenVMS Format

X\$GET\_ERROR\_DATABASE\_TEXT

(display, appl\_name, message\_name, default\_message\_name, buff\_return, [len\_return])

### Argument Information

| Argument             | Usage       | Data Type        | Access | Mechanism  |
|----------------------|-------------|------------------|--------|------------|
| display              | identifier  | uns longword     | read   | reference  |
| appl_name            | char_string | character string | read   | descriptor |
| message_name         | char_string | character string | read   | descriptor |
| default_message_name | char_string | character string | read   | descriptor |
| buff_return          | char_string | character string | write  | descriptor |
| len_return           | word        | uns word         | write  | reference  |

### Arguments

#### **message\_name**

The type of the error message. Xlib uses the following message types to report errors:

| OpenVMS Binding   | Description                                                                                                                                                          |
|-------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| X\$C_PROTO_ERROR  | The protocol error number used as a string for <b>message_name</b> .                                                                                                 |
| X\$C_XLIB_MESSAGE | The message strings used internally by Xlib.                                                                                                                         |
| X\$C_REQUEST      | The major protocol request number used for the message argument. If no string is found in the error database, the default string is returned to <b>buff_return</b> . |

The **message\_name** argument is the address of a character string descriptor that points to the string.

**length**

The size of the buffer that is passed in **buff\_return**. This argument is optional.

## GET ERROR TEXT

### OpenVMS Format

```
X$GET_ERROR_TEXT
    (display, code, buff_return, [len_return])
```

### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| display     | identifier  | uns longword     | read   | reference  |
| code        | longword    | longword         | read   | reference  |
| buff_return | char_string | character string | write  | descriptor |
| len_return  | longword    | uns longword     | write  | reference  |

### Arguments

**buff\_return**

A pointer to the buffer to which GET ERROR TEXT returns the description. The **buff\_return** argument is the address of a character string descriptor that points to the string.

**len\_return**

The size of the text string that GET ERROR TEXT returns. This argument is optional.

## GET FONT PATH

### OpenVMS Format

```
status_return = X$GET_FONT_PATH
    (display, num_paths_return, directories_return, [len_return])
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |

## Xlib Routines

### GET FONT PATH

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| num_paths_return   | longword    | longword         | write  | reference  |
| directories_return | char_string | character string | write  | descriptor |
| len_return         | word        | uns word         | write  | reference  |

### Returns

#### status\_return

Specifies whether the routine completed successfully. GET FONT PATH returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| SS\$_NORMAL    | Routine completed successfully. |
| X\$_NOTFOUND   | Directory list not found.       |
| X\$_ERRORREPLY | Error received from server.     |

### Arguments

#### directories\_return

Comma separated list of directories that comprise the current font directory path.

#### len\_return

The length of the returned string. This argument is optional.

---

## GET FONT PROPERTY

### OpenVMS Format

```
status_return = X$GET_FONT_PROPERTY  
(font_struct, atom_id, value_return)
```

### Argument Information

| Argument      | Usage      | Data Type      | Access | Mechanism |
|---------------|------------|----------------|--------|-----------|
| status_return | longword   | uns longword   | write  | value     |
| font_struct   | record     | x\$font_struct | read   | reference |
| atom_id       | identifier | uns longword   | read   | reference |
| value_return  | longword   | longword       | write  | value     |

### Returns

#### status\_return

Specifies whether the return completed successfully. The routine returns zero for success and a non-zero value for failure.

---

## GET GC VALUES

### OpenVMS Format

```
status_return = X$GET_GC_VALUES
    (display, gc_id, valuemask, values_return)
```

### Argument Information

| Argument      | Usage             | Data Type    | Access | Mechanism |
|---------------|-------------------|--------------|--------|-----------|
| status_return | longword          | uns longword | write  | value     |
| display       | identifier        | uns longword | read   | reference |
| gc_id         | identifier        | uns longword | read   | reference |
| value_mask    | mask_<br>longword | uns longword | read   | reference |
| values_return | record            | x\$gc_values | write  | reference |

### Returns

**status\_return**  
Specifies whether the return completed successfully. The routine returns zero for success and a non-zero value for failure.

---

## GET GEOMETRY

### OpenVMS Format

```
status_return = X$GET_GEOMETRY
    (display, drawable_id, [window_id_return], [x_coord_return], [y_coord_return],
    [width_return], [height_return], [border_width_return], [depth_return])
```

### Argument Information

| Argument         | Usage        | Data Type    | Access | Mechanism |
|------------------|--------------|--------------|--------|-----------|
| status_return    | longword     | longword     | write  | value     |
| display          | identifier   | uns longword | read   | reference |
| drawable_id      | identifier   | uns longword | read   | reference |
| window_id_return | identifier   | uns longword | write  | reference |
| x_coord_return   | longword     | longword     | write  | reference |
| y_coord_return   | longword     | longword     | write  | reference |
| width_return     | uns longword | uns longword | write  | reference |
| height_return    | uns longword | uns longword | write  | reference |

## Xlib Routines

### GET GEOMETRY

| Argument                | Usage        | Data Type    | Access | Mechanism |
|-------------------------|--------------|--------------|--------|-----------|
| border_width_<br>return | uns longword | uns longword | write  | reference |
| depth_return            | longword     | longword     | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. GET GEOMETRY returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| X\$_ERRORREPLY | Error received from server.     |
| SS\$_NORMAL    | Routine completed successfully. |

### Arguments

#### window\_id\_return

The identifier of the root window of the specified drawable. This argument is optional.

#### x\_coord\_return

If the drawable is a window, **x\_coord\_return** specifies the x-coordinate of the window. The x- and y-coordinates define the upper left corner of the window, relative to the origin of the parent window. If the drawable is a pixmap, **x\_coord\_return** and **y\_coord\_return** equal zero. This argument is optional.

#### y\_coord\_return

If the drawable is a window, **y\_coord\_return** specifies the y-coordinate of the window. The x- and y-coordinates define the upper left corner of the window, relative to the origin of the parent window. If the drawable is a pixmap, **x\_coord\_return** and **y\_coord\_return** equal zero. This argument is optional.

#### width\_return

The width, in pixels, of the drawable. GET GEOMETRY returns the width of the drawable to this argument. The width and height represent the inside area of the drawable, not including the border of the drawable. This argument is optional.

#### height\_return

The height, in pixels, of the drawable. GET GEOMETRY returns the height of the drawable to this argument. The width and height represent the inside area of the drawable, not including the border of the drawable. This argument is optional.

#### border\_width\_return

The width, in pixels, of the new subwindow's border. If the drawable is a window, GET GEOMETRY returns the width of the window's border to this argument. If the drawable is a pixmap, GET GEOMETRY returns zero to this argument. This argument is optional.



**depth\_return**

The depth of the pixmap. The depth must be supported by the root window of the specified drawable. GET GEOMETRY returns the depth, in bits per pixel, to this argument. This argument is optional.

## GET ICON NAME

### OpenVMS Format

```
status_return = X$GET_ICON_NAME
    (display, window_id, icon_name_return, [icon_len_return])
```

### Argument Information

| Argument         | Usage       | Data Type        | Access | Mechanism  |
|------------------|-------------|------------------|--------|------------|
| status_return    | cond_value  | uns longword     | write  | value      |
| display          | identifier  | uns longword     | read   | reference  |
| window_id        | identifier  | uns longword     | read   | reference  |
| icon_name_return | char_string | character string | write  | descriptor |
| icon_len_return  | word        | uns word         | write  | reference  |

### Returns

**status\_return**

Specifies whether the routine completed successfully. GET ICON NAME returns one of the following values:

| Value           | Description                                               |
|-----------------|-----------------------------------------------------------|
| X\$_ERRORREPLY  | Error received from server.                               |
| X\$_TRUNCATED   | Results truncated; user-supplied buffer not large enough. |
| LIB\$_STRTRU    | Font names returned but truncated.                        |
| LIB\$_FATALERR  | LIB\$SCOPY_R_DX fatal error.                              |
| LIB\$_INSVIRMEM | Insufficient virtual memory; LIB\$GET_VM call failed.     |
| LIB\$_INVSTRDES | Invalid string descriptor.                                |
| SS\$_NORMAL     | Routine completed successfully.                           |
| 0               | Indicates an error.                                       |

**icon\_len\_return**

Length of the icon name. This argument is optional.

---

## GET ICON SIZES

### OpenVMS Format

```
status_return = X$GET_ICON_SIZES  
(display, window_id, [size_list_return], [count_return], [list_size], [list_buff_return])
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| status_return    | cond_value | uns longword | write  | value     |
| display          | identifier | uns longword | read   | reference |
| window_id        | identifier | uns longword | read   | reference |
| size_list_return | address    | uns longword | write  | reference |
| count_return     | longword   | longword     | write  | reference |
| list_size        | longword   | longword     | read   | reference |
| list_buff_return | array      | uns longword | write  | reference |

### Arguments

#### **size\_list\_return**

The virtual address of a pointer to an array of icon size data, returned by the routine and residing in space reserved by Xlib. The recommended icon size is defined by minimum, maximum, and incremental width and height specifications. If the incremental width and height specifications are zero, then a single size is recommended. If the incremental specifications are nonzero, then the minimum size plus an increment up to the maximum size is permitted. This argument is optional.

#### **count\_return**

The number of items in **size\_list\_return**. This argument is optional.

#### **list\_size**

The size of the buffer in **list\_buff\_return**. This argument is optional.

#### **list\_buff\_return**

A pointer to a data buffer residing in space you have reserved, where each entry is one icon size data element. The size of the buffer is specified by **list\_size**. The icon size data is returned by the routine. This argument is optional.

---

## GET IMAGE

### OpenVMS Format

status\_return = X\$GET\_IMAGE

(display, drawable\_id, x\_coord, y\_coord, width, height, plane\_mask, image\_format, ximage\_return)

### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| status_return | longword      | longword     | write  | value     |
| display       | identifier    | uns longword | read   | reference |
| drawable_id   | identifier    | uns longword | read   | reference |
| x_coord       | longword      | longword     | read   | reference |
| y_coord       | longword      | longword     | read   | reference |
| width         | longword      | uns longword | read   | reference |
| height        | longword      | uns longword | read   | reference |
| plane_mask    | mask_longword | uns longword | read   | reference |
| image_format  | longword      | longword     | read   | reference |
| ximage_return | record        | x\$image     | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. GET IMAGE returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| SSS_NORMAL     | Routine completed successfully. |
| X\$_ERRORREPLY | Error received from server.     |

### Arguments

#### image\_format

The format of the image. The predefined values for **image\_format** are as follows:

XSC\_XY\_BITMAP  
XSC\_XY\_PIXMAP  
XSC\_Z\_PIXMAP

Other values specified in this argument are not valid.

#### ximage\_return

The returned image data structure containing the requested data.

---

## GET INPUT FOCUS

### OpenVMS Format

X\$GET\_INPUT\_FOCUS  
(display, [focus\_id\_return], [revert\_to\_return])

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| display          | identifier | uns longword | read   | reference |
| focus_id_return  | identifier | uns longword | write  | value     |
| revert_to_return | longword   | longword     | write  | value     |

### Arguments

#### focus\_id\_return

The identifier of the focus window, Pointer Root, or None. The identifier of the window was originally returned by CREATE SIMPLE WINDOW or CREATE WINDOW. This argument is optional.

#### revert\_to\_return

The current input focus state. One of the following predefined values can be returned:

X\$C\_REVERT\_TO\_PARENT  
X\$C\_REVERT\_TO\_POINTER\_ROOT  
X\$C\_REVERT\_TO\_NONE

This argument is optional.

---

## GET KEYBOARD CONTROL

### OpenVMS Format

X\$GET\_KEYBOARD\_CONTROL  
(display, state\_values\_return)

### Argument Information

| Argument            | Usage      | Data Type         | Access | Mechanism |
|---------------------|------------|-------------------|--------|-----------|
| display             | identifier | uns longword      | read   | reference |
| state_values_return | record     | x\$keyboard_state | write  | reference |

## GET KEYBOARD MAPPING

### OpenVMS Format

```
status_return = X$GET_KEYBOARD_MAPPING
    (display, first_keycode_wanted, keycode_count, [keysyms_per_keycode_return],
    [keysyms_return], [buff_size], [key_buff_return])
```

### Argument Information

| Argument                   | Usage      | Data Type    | Access | Mechanism |
|----------------------------|------------|--------------|--------|-----------|
| status_return              | cond_value | uns longword | write  | value     |
| display                    | identifier | uns longword | read   | reference |
| first_keycode_wanted       | identifier | uns longword | read   | reference |
| keycode_count              | longword   | longword     | read   | reference |
| keysyms_per_keycode_return | longword   | longword     | write  | reference |
| keysyms_return             | address    | uns longword | write  | reference |
| buff_size                  | longword   | longword     | read   | reference |
| key_buff_return            | array      | uns longword | write  | reference |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

### Arguments

**keysyms\_return**  
The virtual address of the symbol list returned by the routine and residing in space reserved by Xlib. If you specify this optional argument, GET KEYBOARD MAPPING determines the size of the buffer to create for the symbol list. If you specify **keysyms\_return**, you do not need to specify **buff\_size** and **key\_buff\_return**.

**buff\_size**  
The size of the **key\_buff\_return** buffer. This argument is optional.

**key\_buff\_return**  
A pointer to an array residing in space you have reserved, in which each element is a key symbol. GET KEYBOARD MAPPING returns the key symbols to this array. This argument is optional.

## Xlib Routines

### GET MODIFIER MAPPING

---

## GET MODIFIER MAPPING

### OpenVMS Format

X\$GET\_MODIFIER\_MAPPING  
(display, modifier\_keys\_return)

### Argument Information

| Argument             | Usage      | Data Type          | Access | Mechanism |
|----------------------|------------|--------------------|--------|-----------|
| display              | identifier | uns longword       | read   | reference |
| modifier_keys_return | record     | x\$modifier_keymap | write  | reference |

### Arguments

#### **modifier\_keys\_return**

The modifier key map data structure containing the values for the modifier keys. GET MODIFIER MAPPING returns the values in this argument.

---

## GET MOTION EVENTS

### OpenVMS Format

status\_return = X\$GET\_MOTION\_EVENTS  
(display, window\_id, start, stop, num\_events\_return, [time\_return], [time\_size], [time\_buff\_return])

### Argument Information

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| status_return     | cond_value | uns longword | write  | value     |
| display           | identifier | uns longword | read   | reference |
| window_id         | identifier | uns longword | read   | reference |
| start             | longword   | uns longword | read   | reference |
| stop              | longword   | uns longword | read   | reference |
| num_events_return | longword   | longword     | write  | reference |
| time_return       | address    | uns longword | write  | reference |
| time_size         | longword   | longword     | read   | reference |
| time_buff_return  | array      | uns longword | write  | reference |

## Returns

### **status\_return**

Specifies whether the routine completed successfully. GET MOTION EVENTS returns one of the following values:

| Value         | Description                                                        |
|---------------|--------------------------------------------------------------------|
| XS_ERRORREPLY | Error received from server.                                        |
| XS_TRUNCATED  | User buffer specified in <b>time_buff_return</b> not large enough. |
| SSS_NORMAL    | Routine completed successfully.                                    |

## Arguments

### **time\_return**

The address of the time coordinate data structure returned by the routine and residing in space reserved by Xlib. You can specify either **time\_return** to receive the address of the time coordinate data structure, or the **time\_size** and **time\_buff\_return** arguments to receive an array of time coordinates in a buffer. This argument is optional.

### **time\_size**

The length of the time coordinate buffer. This argument is optional.

### **time\_buff\_return**

A pointer to an array of time coordinates. The length of the array is specified by **time\_size** and resides in space that you have reserved. This argument is optional.

---

## GET NORMAL HINTS

### OpenVMS Format

```
status_return = X$GET_NORMAL_HINTS
    (display, window_id, hints_return)
```

### Argument Information

| Argument      | Usage      | Data Type     | Access | Mechanism |
|---------------|------------|---------------|--------|-----------|
| status_return | longword   | uns longword  | write  | value     |
| display       | identifier | uns longword  | read   | reference |
| window_id     | identifier | uns longword  | read   | reference |
| hints_return  | record     | x\$size_hints | write  | reference |

## Xlib Routines

### GET NORMAL HINTS

#### Returns

##### **status\_return**

Specifies whether the return completed successfully. The routine returns a non-zero value for success and zero for failure.

---

## GET PIXEL

#### OpenVMS Format

```
pixel_value_return = X$GET_PIXEL  
(ximage, x_coord, y_coord)
```

#### Argument Information

| Argument           | Usage    | Data Type    | Access | Mechanism |
|--------------------|----------|--------------|--------|-----------|
| pixel_value_return | longword | uns longword | write  | reference |
| ximage             | record   | x\$image     | read   | reference |
| x_coord            | longword | longword     | read   | reference |
| y_coord            | longword | longword     | read   | reference |

---

## GET POINTER CONTROL

#### OpenVMS Format

```
X$GET_POINTER_CONTROL  
(display, [accel_numerator_return], [accel_denominator_return], [threshold_return])
```

#### Argument Information

| Argument                 | Usage      | Data Type    | Access | Mechanism |
|--------------------------|------------|--------------|--------|-----------|
| display                  | identifier | uns longword | read   | reference |
| accel_numerator_return   | longword   | longword     | write  | reference |
| accel_denominator_return | longword   | longword     | write  | reference |
| threshold_return         | longword   | longword     | write  | reference |

#### Arguments

##### **accel\_numerator\_return**

The acceleration numerator. The **accel\_numerator\_return** and **accel\_denominator\_return** arguments specify the complete acceleration multiplier. This argument is optional.



**accel\_denominator\_return**

The acceleration denominator. The **accel\_numerator\_return** and **accel\_denominator\_return** arguments specify the complete acceleration multiplier. This argument is optional.

**threshold\_return**

The acceleration threshold, specified in the number of pixels moved during one movement. This argument is optional.

---

## GET POINTER MAPPING

### OpenVMS Format

```
num_elements_return = X$GET_POINTER_MAPPING  
(display, map_return, num_map)
```

## Xlib Routines

### GET POINTER MAPPING

#### Argument Information

| Argument            | Usage      | Data Type    | Access | Mechanism |
|---------------------|------------|--------------|--------|-----------|
| num_elements_return | longword   | longword     | write  | value     |
| display             | identifier | uns longword | read   | reference |
| map_return          | array      | byte         | write  | reference |
| num_map             | word       | uns word     | read   | reference |

---

## GET RGB COLORMAPS

#### OpenVMS Format

status\_return = X\$GET\_RGB\_COLORMAPS

(display, window\_id, [standard\_colormap\_return], [count\_return], property\_id,  
[colormap\_size], [colormap\_buff\_return])

#### Argument Information

| Argument                     | Usage        | Data Type    | Access | Mechanism |
|------------------------------|--------------|--------------|--------|-----------|
| status_return                | cond_value   | uns longword | write  | value     |
| display                      | identifer    | uns longword | read   | reference |
| window_id                    | identifier   | uns longword | read   | reference |
| standard_<br>colormap_return | address      | array        | write  | reference |
| count_return                 | uns longword | uns longword | write  | reference |
| property_id                  | identifier   | uns longword | read   | reference |
| colormap_size                | uns longword | uns longword | read   | reference |
| colormap_buff_<br>return     | array        | uns longword | write  | reference |

#### Arguments

##### **colormap\_size**

Size of the **colormap\_buff\_return** argument in bytes. This argument is optional.

##### **colormap\_buff\_return**

Pointer to an array residing in space reserved you have reserved. Each entry is a window ID. This argument is optional.

---

## GET SCREEN SAVER

### OpenVMS Format

```
X$GET_SCREEN_SAVER
    (display, [timeout_return], [interval_return], [prefer_blanking_return],
    [allow_exposures_return])
```

### Argument Information

| Argument               | Usage      | Data Type    | Access | Mechanism |
|------------------------|------------|--------------|--------|-----------|
| display                | identifier | uns longword | read   | reference |
| timeout_return         | longword   | longword     | write  | reference |
| interval_return        | longword   | longword     | write  | reference |
| prefer_blanking_return | longword   | longword     | write  | reference |
| allow_exposures_return | longword   | longword     | write  | reference |

### Arguments

#### **timeout\_return**

The time, in seconds, that the screen saver waits before turning on. The time represents the number of seconds when no input from the keyboard or pointing device is received. A value of zero means that the screen saver is disabled. This argument is optional.

#### **interval\_return**

The time, in seconds, from one screen saver invocation to the next. This argument is optional.

#### **prefer\_blanking\_return**

The screen blanking mode. The predefined values for **prefer\_blanking\_return** are as follows:

```
X$C_DONT_PREFER_BLANKING
X$C_PREFER_BLANKING
X$C_DEFAULT_BLANKING
```

This argument is optional.

#### **allow\_exposures\_return**

The current screen saver control values are returned. The predefined values for **allow\_exposures\_return** are as follows:

```
X$C_DONT_ALLOW_EXPOSURES
X$C_ALLOW_EXPOSURES
X$C_DEFAULT_EXPOSURES
```

This argument is optional.

## Xlib Routines

### GET SELECTION OWNER

---

## GET SELECTION OWNER

### OpenVMS Format

owner\_id\_return = X\$GET\_SELECTION\_OWNER  
(display, selection\_id)

### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| owner_id_return | identifier | uns longword | write  | value     |
| display         | identifier | uns longword | read   | reference |
| selection_id    | identifier | uns longword | read   | reference |

---

## GET SIZE HINTS

### OpenVMS Format

status\_return = X\$GET\_SIZE\_HINTS  
(display, window\_id, hints\_return, property\_id)

### Argument Information

| Argument      | Usage      | Data Type     | Access | Mechanism |
|---------------|------------|---------------|--------|-----------|
| status_return | cond_value | uns longword  | write  | value     |
| display       | identifier | uns longword  | read   | reference |
| window_id     | identifier | uns longword  | read   | reference |
| hints_return  | record     | x\$size_hints | write  | reference |
| property_id   | identifier | uns longword  | read   | reference |

---

## GET STANDARD COLORMAP

### OpenVMS Format

status\_return = X\$GET\_STANDARD\_COLORMAP  
(display, window\_id, standard\_colormap\_return, property\_id)

**Argument Information**

| Argument                 | Usage      | Data Type            | Access | Mechanism |
|--------------------------|------------|----------------------|--------|-----------|
| status_return            | cond_value | uns longword         | write  | value     |
| display                  | identifier | uns longword         | read   | reference |
| window_id                | identifier | uns longword         | read   | reference |
| standard_colormap_return | record     | x\$standard_colormap | write  | reference |
| property_id              | identifier | uns longword         | read   | reference |

**Returns**

**status\_return**

Specifies whether the routine completed successfully. The routine fails if the standard color map property is not defined for the specified window. This argument returns one of the following values.

| Value          | Description                                                                    |
|----------------|--------------------------------------------------------------------------------|
| SS\$_NORMAL    | Routine completed successfully.                                                |
| X\$_PROPUNDEF  | The standard color map property has not been defined for the specified window. |
| X\$_ERRORREPLY | Error received from server.                                                    |
| X\$_TRUNCATED  | Buffer not large enough; results truncated.                                    |

---

**GET SUB IMAGE**

**OpenVMS Format**

status\_return = X\$GET\_SUB\_IMAGE

(display, drawable\_id, x\_coord, y\_coord, width, height, plane\_mask, image\_format, dst\_ximage\_return, dst\_x\_coord, dst\_y\_coord)

**Argument Information**

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| drawable_id   | identifier | uns longword | read   | reference |
| x_coord       | longword   | longword     | read   | reference |
| y_coord       | longword   | longword     | read   | reference |
| width         | longword   | uns longword | read   | reference |

## Xlib Routines

### GET SUB IMAGE

| Argument          | Usage         | Data Type    | Access | Mechanism |
|-------------------|---------------|--------------|--------|-----------|
| height            | longword      | uns longword | read   | reference |
| plane_mask        | mask_longword | uns longword | read   | reference |
| image_format      | longword      | longword     | read   | reference |
| dst_ximage_return | record        | x\$image     | write  | reference |
| dst_x_coord       | longword      | longword     | read   | reference |
| dst_y_coord       | longword      | longword     | read   | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. GET SUB IMAGE returns one of the following values:

| Value          | Description                     |
|----------------|---------------------------------|
| SS\$_NORMAL    | Routine completed successfully. |
| X\$_ERRORREPLY | Error received from server.     |

### Arguments

#### image\_format

The format of the image. The predefined values for **image\_format** are as follows:

```
XSC_XY_BITMAP
XSC_XY_PIXMAP
XSC_Z_PIXMAP
```

Other values specified in this argument are not valid.

---

## GET TEXT PROPERTY

### OpenVMS Format

```
status_return = X$GET_TEXT_PROPERTY
```

```
(display, window_id, text_prop_return, property_id)
```

### Argument Information

| Argument         | Usage      | Data Type        | Access | Mechanism |
|------------------|------------|------------------|--------|-----------|
| status_return    | longword   | uns longword     | write  | value     |
| display          | identifier | uns longword     | read   | reference |
| window_id        | identifer  | uns longword     | read   | reference |
| text_prop_return | record     | x\$text_property | write  | reference |
| property_id      | identifier | uns longword     | read   | reference |

## Returns

### **status\_return**

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## GET TRANSIENT FOR HINT

### OpenVMS Format

```
status_return = X$GET_TRANSIENT_FOR_HINT
    (display, window_id, prop_window_return)
```

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| status_return      | cond_value | uns longword | write  | value     |
| display            | identifier | uns longword | read   | reference |
| window_id          | identifier | uns longword | read   | reference |
| prop_window_return | identifier | uns longword | read   | reference |

---

## GET VISUAL INFO

### OpenVMS Format

```
status_return = X$GET_VISUAL_INFO
    (display, vinfo_mask, vinfo_template, num_items_return, [items_return] [items_size],
    [items_buff_return])
```

### Argument Information

| Argument         | Usage         | Data Type      | Access | Mechanism |
|------------------|---------------|----------------|--------|-----------|
| status_return    | cond_value    | uns longword   | write  | value     |
| display          | identifier    | uns longword   | read   | reference |
| vinfo_mask       | mask_longword | uns longword   | read   | reference |
| vinfo_template   | record        | x\$visual_info | read   | reference |
| num_items_return | longword      | longword       | write  | reference |

## Xlib Routines

### GET VISUAL INFO

| Argument          | Usage    | Data Type    | Access | Mechanism |
|-------------------|----------|--------------|--------|-----------|
| items_return      | address  | uns longword | write  | reference |
| items_size        | longword | longword     | read   | reference |
| items_buff_return | record   | uns longword | read   | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. GET VISUAL INFO returns one of the following values:

| Value         | Description                                 |
|---------------|---------------------------------------------|
| SS\$_NORMAL   | Routine completed successfully.             |
| X\$_TRUNCATED | Buffer not large enough; results truncated. |

### Returns

#### items\_return

The virtual address of a pointer to an array of visual information data returned by the routine and residing in space reserved by Xlib.

#### items\_size

The size of the buffer specified in **items\_buff\_return**.

#### items\_buff\_return

A pointer to a data buffer, residing in space you have reserved, where each entry is one visual information element. The size of the buffer is specified by **items\_size**.

---

## GET WINDOW ATTRIBUTES

### OpenVMS Format

```
status_return = X$GET_WINDOW_ATTRIBUTES
```

```
(display, window_id, window_attributes_return)
```

### Argument Information

| Argument                 | Usage      | Data Type            | Access | Mechanism |
|--------------------------|------------|----------------------|--------|-----------|
| status_return            | cond_value | uns longword         | write  | value     |
| display                  | identifier | uns longword         | read   | reference |
| window_id                | identifier | uns longword         | read   | reference |
| window_attributes_return | array      | x\$window_attributes | write  | reference |



## Returns

### **status\_return**

Specifies whether the routine completed successfully.

This argument returns one of the following values:

| Value          | Description                                          |
|----------------|------------------------------------------------------|
| X\$_ERRORREPLY | Error received from server; window no longer exists. |
| SS\$NORMAL     | Routine completed successfully.                      |
| Otherwise      | Failure for reason given.                            |

---

## GET WINDOW PROPERTY

### OpenVMS Format

status\_return = X\$GET\_WINDOW\_PROPERTY

(display, window\_id, property\_id, long\_offset, long\_len, delete, requested\_type, actual\_type\_return, actual\_format\_return, num\_elements\_return, bytes\_after\_return, [property\_data\_return], [property\_data\_len], [property\_data\_buff\_return])

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| status_return        | cond_value | uns longword | write  | value     |
| display              | identifier | uns longword | read   | reference |
| window_id            | identifier | uns longword | read   | reference |
| property_id          | identifier | uns longword | read   | reference |
| long_offset          | longword   | longword     | read   | reference |
| long_len             | longword   | longword     | read   | reference |
| delete               | longword   | uns longword | read   | reference |
| requested_type       | identifier | uns longword | read   | reference |
| actual_type_return   | identifier | uns longword | write  | reference |
| actual_format_return | longword   | longword     | write  | reference |
| num_elements_return  | longword   | longword     | write  | reference |
| bytes_after_return   | longword   | longword     | write  | reference |

## Xlib Routines

### GET WINDOW PROPERTY

| Argument                  | Usage    | Data Type    | Access | Mechanism |
|---------------------------|----------|--------------|--------|-----------|
| property_data_return      | address  | uns longword | write  | reference |
| property_data_len         | longword | longword     | read   | reference |
| property_data_buff_return | array    | uns longword | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. This argument returns a zero if the routine completes successfully, and one of the following values if it does not complete successfully:

| Value          | Description                                               |
|----------------|-----------------------------------------------------------|
| X\$_ERRORREPLY | Error received from server.                               |
| X\$_TRUNCATED  | Results truncated; user-supplied buffer not large enough. |

### Arguments

#### property\_data\_return

Virtual address of a pointer to an array of property data returned by the routine and residing in space reserved by Xlib.

#### property\_data\_len

The size of the data buffer specified in **property\_data\_buff\_return**. This argument is optional.

#### property\_data\_buff\_return

A pointer to a data buffer residing in space you reserved, where each entry is one property element. The length of the buffer is specified by **property\_data\_len**. The property data is returned by the routine. This argument is optional.

---

## GET WM CLIENT MACHINE

### OpenVMS Format

```
status_return = X$GET_WM_CLIENT_MACHINE  
(display, window_id, property_return)
```

### Argument Information

| Argument      | Usage    | Data Type    | Access | Mechanism |
|---------------|----------|--------------|--------|-----------|
| status_return | longword | uns longword | write  | value     |

## Xlib Routines GET WM CLIENT MACHINE

| Argument        | Usage      | Data Type        | Access | Mechanism |
|-----------------|------------|------------------|--------|-----------|
| display         | identifier | uns longword     | read   | reference |
| window_id       | identifer  | uns longword     | read   | reference |
| property_return | record     | x\$text_property | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## GET WM COLORMAP WINDOWS

### OpenVMS Format

```
status_return = X$GET_WM_COLORMAP_WINDOWS
```

```
(display, window_id, [colormap_windows_return], [count_return], [colormap_size],
[colormap_buff_return])
```

### Argument Information

| Argument                    | Usage      | Data Type    | Access | Mechanism |
|-----------------------------|------------|--------------|--------|-----------|
| status_return               | cond_value | uns longword | write  | value     |
| display                     | identifier | uns longword | read   | reference |
| window_id                   | identifier | uns longword | read   | reference |
| colormap_<br>windows_return | address    | array        | write  | reference |
| count_return                | longword   | uns longword | write  | reference |
| colormap_size               | longword   | uns longword | read   | reference |
| colormap_buff_<br>return    | array      | uns longword | write  | reference |

### Arguments

#### colormap\_windows\_return

List of identifiers stored in the WM\_COLORMAP\_WINDOWS property of the specified window. The list is returned by the routine and resides in space reserved by Xlib. This argument is optional.

#### count\_return

Number of windows in the list. This argument is optional.

## Xlib Routines

### GET WM COLORMAP WINDOWS

#### **colormap\_size**

Size of the **colormap\_buff\_return** argument in bytes. This argument is optional.

#### **colormap\_buff\_return**

Pointer to a data buffer residing in space you have reserved, where each entry is a window ID. This argument is optional.

---

## GET WM HINTS

### OpenVMS Format

X\$GET\_WM\_HINTS

(display, window\_id, wmhints\_return)

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| display        | identifier | uns longword | read   | reference |
| window_id      | identifier | uns longword | read   | reference |
| wmhints_return | cond_value | x\$wm_hints  | write  | reference |

### Arguments

#### **wmhints\_return**

The window manager hints data structure returned by the routine.

---

## GET WM ICON NAME

### OpenVMS Format

status\_return = X\$GET\_WM\_ICON\_NAME

(display, window\_id, property\_return)

### Argument Information

| Argument        | Usage      | Data Type        | Access | Mechanism |
|-----------------|------------|------------------|--------|-----------|
| status_return   | longword   | uns longword     | write  | value     |
| display         | identifier | uns longword     | read   | reference |
| window_id       | identifier | uns longword     | read   | reference |
| property_return | record     | x\$text_property | write  | reference |

## Returns

### **status\_return**

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## GET WM NAME

### OpenVMS Format

```
status_return = X$GET_WM_NAME
    (display, window_id, property_return)
```

### Argument Information

| Argument        | Usage      | Data Type        | Access | Mechanism |
|-----------------|------------|------------------|--------|-----------|
| status_return   | longword   | uns longword     | write  | value     |
| display         | identifier | uns longword     | read   | reference |
| window_id       | identifier | uns longword     | read   | reference |
| property_return | record     | x\$text_property | write  | reference |

## Returns

### **status\_return**

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## GET WM NORMAL HINTS

### OpenVMS Format

```
status_return = X$GET_WM_NORMAL_HINTS
    (display, window_id, hints_return, supplied_return)
```

## Xlib Routines

### GET WM NORMAL HINTS

#### Argument Information

| Argument        | Usage             | Data Type     | Access | Mechanism |
|-----------------|-------------------|---------------|--------|-----------|
| status_return   | longword          | uns longword  | write  | value     |
| display         | identifier        | uns longword  | read   | reference |
| window_id       | identifier        | uns longword  | read   | reference |
| hints_return    | record            | x\$size_hints | write  | reference |
| supplied_return | mask_<br>longword | uns longword  | write  | reference |

#### Returns

##### status\_return

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## GET WM PROTOCOLS

#### OpenVMS Format

status\_return = X\$GET\_WM\_PROTOCOLS

(display, window\_id, [protocols\_return], [count\_return], [prot\_size],  
[prot\_buff\_return])

#### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| status_return    | cond_value | uns longword | write  | reference |
| display          | identifier | uns longword | read   | reference |
| window_id        | identifier | uns longword | read   | reference |
| protocols_return | address    | uns longword | write  | reference |
| count_return     | longword   | longword     | write  | reference |
| prot_size        | longword   | longword     | read   | reference |
| prot_buff_return | address    | uns longword | write  | reference |

## Arguments

### **protocols\_return**

A pointer to a data buffer returned by the routine and residing in space reserved by Xlib. Each entry is one protocol element. This argument is optional.

### **prot\_size**

The length of the data buffer specified in **prot\_buff\_return**. This argument is optional.

### **prot\_buff\_return**

A pointer to a data buffer residing in space you have reserved, where each entry is one protocol element. This argument is optional.

## GET ZOOM HINTS

### OpenVMS Format

```
status_return = X$GET_ZOOM_HINTS
    (display, window_id, zhints_return)
```

### Argument Information

| Argument      | Usage      | Data Type     | Access | Mechanism |
|---------------|------------|---------------|--------|-----------|
| status_return | longword   | uns longword  | write  | value     |
| display       | identifier | uns longword  | read   | reference |
| window_id     | identifier | uns longword  | read   | reference |
| zhints_return | record     | x\$size_hints | write  | reference |

## Returns

### **status\_return**

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## GRAB BUTTON

### OpenVMS Format

X\$GRAB\_BUTTON

(display, button, modifiers, window\_id, owner\_events, event\_mask, pointer\_mode, keyboard\_mode, confine\_id, cursor\_id)

### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| display       | identifier    | uns longword | read   | reference |
| button        | longword      | longword     | read   | reference |
| modifiers     | mask_longword | uns longword | read   | reference |
| window_id     | identifier    | uns longword | read   | reference |
| owner_events  | Boolean       | uns longword | read   | reference |
| event_mask    | mask_longword | uns longword | read   | reference |
| pointer_mode  | longword      | longword     | read   | reference |
| keyboard_mode | longword      | longword     | read   | reference |
| confine_id    | identifier    | uns longword | read   | reference |
| cursor_id     | identifier    | uns longword | read   | reference |

### Arguments

#### button

The button on the pointing device to grab when the specified modifier keys are down. The possible predefined values are as follows:

X\$C\_BUTTON1  
X\$C\_BUTTON2  
X\$C\_BUTTON3  
X\$C\_BUTTON4  
X\$C\_BUTTON5  
X\$C\_ANY\_BUTTON

Other buttons pressed are not grabbed. Specify the predefined value X\$C\_ANY\_BUTTON to grab all possible buttons.

#### modifiers

The set of key masks. This mask is the inclusive OR of the following key mask bits:

| Bit | Predefined Value |
|-----|------------------|
| 1   | X\$M_SHIFT       |
| 2   | X\$M_CAPS_LOCK   |
| 3   | X\$M_CONTROL     |



| Bit | Predefined Value |
|-----|------------------|
| 4   | XSM_MOD1         |
| 5   | XSM_MOD2         |
| 6   | XSM_MOD3         |
| 7   | XSM_MOD4         |
| 8   | XSM_MOD5         |

The predefined value XSC\_ANY\_MODIFIER can be specified to allow any set of modifiers to be grabbed.

**event\_mask**

A bit mask that specifies which pointer events are reported to the client.

Table 2–4 lists the predefined values for the **event\_mask**.

**Table 2–4 Event Mask Description**

| Bit | Predefined Value        | Description                          |
|-----|-------------------------|--------------------------------------|
| 2   | XSM_BUTTON_PRESS        | Pointer button down events wanted    |
| 3   | XSM_BUTTON_RELEASE      | Pointer button up events wanted      |
| 4   | XSM_ENTER_WINDOW        | Pointer window entry events wanted   |
| 5   | XSM_LEAVE_WINDOW        | Pointer window leave events wanted   |
| 6   | XSM_POINTER_MOTION      | Pointer motion events wanted         |
| 7   | XSM_POINTER_MOTION_HINT | Pointer motion hints wanted          |
| 8   | XSM_BUTTON1_MOTION      | Pointer motion while button 1 down   |
| 9   | XSM_BUTTON2_MOTION      | Pointer motion while button 2 down   |
| 10  | XSM_BUTTON3_MOTION      | Pointer motion while button 3 down   |
| 11  | XSM_BUTTON4_MOTION      | Pointer while button 4 down          |
| 12  | XSM_BUTTON5_MOTION      | Pointer motion while button 5 down   |
| 13  | XSM_BUTTON_MOTION       | Pointer motion while any button down |
| 14  | XSM_KEYMAP_STATE        | Any keyboard state change wanted     |

**pointer\_mode**

A constant that controls further processing of pointer events. Clients can pass one of the following constants:

XSC\_GRAB\_MODE\_SYNC  
XSC\_GRAB\_MODE\_ASYNC

Other values specified in this argument are not valid.

**keyboard\_mode**

The mode that the keyboard events will use. The predefined values for **keyboard\_mode** are as follows:

XSC\_GRAB\_MODE\_SYNC  
XSC\_GRAB\_MODE\_ASYNC

---

## GRAB KEY

### OpenVMS Format

X\$GRAB\_KEY

(display, keycode, modifiers, window\_id, owner\_events, pointer\_mode, keyboard\_mode)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| display       | identifier | uns longword | read   | reference |
| keycode       | longword   | longword     | read   | reference |
| modifiers     | longword   | uns longword | read   | reference |
| window_id     | identifier | uns longword | read   | reference |
| owner_events  | Boolean    | longword     | read   | reference |
| pointer_mode  | longword   | longword     | read   | reference |
| keyboard_mode | longword   | longword     | read   | reference |

### Arguments

#### modifiers

A bit mask that specifies the set of key masks. This mask is the inclusive OR of the following key mask bits:

| Bit | Predefined Value |
|-----|------------------|
| 1   | X\$M_SHIFT       |
| 2   | X\$M_CAPS_LOCK   |
| 3   | X\$M_CONTROL     |
| 4   | X\$M_MOD1        |
| 5   | X\$M_MOD2        |
| 6   | X\$M_MOD3        |
| 7   | X\$M_MOD4        |
| 8   | X\$M_MOD5        |

The predefined value X\$C\_ANY\_MODIFIER can be specified to allow any set of modifiers to be grabbed, including the combination of no modifiers.

#### pointer\_mode

The processing of pointer events. The predefined values for **pointer\_mode** are as follows:

X\$C\_GRAB\_MODE\_SYNC  
X\$C\_GRAB\_MODE\_ASYNC

Other values specified in this argument are not valid.

**keyboard\_mode**

The processing of keyboard events. The predefined values for **keyboard\_mode** are as follows:

X\$C\_GRAB\_MODE\_SYNC  
X\$C\_GRAB\_MODE\_ASYNC

## GRAB KEYBOARD

### OpenVMS Format

status\_return = X\$GRAB\_KEYBOARD

(display, window\_id, owner\_events, pointer\_mode, keyboard\_mode, time)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | longword   | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| window_id     | identifier | uns longword | read   | reference |
| owner_events  | Boolean    | longword     | read   | reference |
| pointer_mode  | longword   | longword     | read   | reference |
| keyboard_mode | longword   | longword     | read   | reference |
| time          | longword   | uns longword | read   | reference |

### Returns

**status\_return**

Specifies whether the routine completed successfully. GRAB KEYBOARD returns one of the following status values:

| Value                  | Description                                                                                                      |
|------------------------|------------------------------------------------------------------------------------------------------------------|
| X\$C_GRAB_SUCCESS      | Routine completed successfully.                                                                                  |
| X\$C_ALREADY_GRABBED   | Keyboard is actively grabbed by another client.                                                                  |
| X\$C_GRAB_FROZEN       | Keyboard is frozen by an active grab of another client.                                                          |
| X\$C_GRAB_INVALID_TIME | Time specified in <b>time</b> is earlier than the last pointer grab time, or later than the current server time. |
| X\$C_GRAB_NOT_VIEWABLE | Windows specified in <b>window_id</b> are not currently viewable.                                                |

## Xlib Routines

### GRAB\_KEYBOARD

#### Arguments

##### **pointer\_mode**

The constant that specifies the processing of pointer events. The predefined values for **pointer\_mode** are as follows:

XSC\_GRAB\_MODE\_SYNC  
XSC\_GRAB\_MODE\_ASYNC

##### **keyboard\_mode**

The processing of keyboard events. The predefined values for **keyboard\_mode** are as follows:

XSC\_GRAB\_MODE\_SYNC  
XSC\_GRAB\_MODE\_ASYNC

---

## GRAB\_POINTER

#### OpenVMS Format

status\_return = X\$GRAB\_POINTER

(display, window\_id, owner\_events, event\_mask, pointer\_mode, keyboard\_mode, confine\_id, cursor\_id, time)

#### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| status_return | cond_value    | uns longword | write  | value     |
| display       | identifier    | uns longword | read   | reference |
| window_id     | identifier    | uns longword | read   | reference |
| owner_events  | Boolean       | longword     | read   | reference |
| event_mask    | mask_longword | uns longword | read   | reference |
| pointer_mode  | longword      | longword     | read   | reference |
| keyboard_mode | longword      | longword     | read   | reference |
| confine_id    | identifier    | uns longword | read   | reference |
| cursor_id     | identifier    | uns longword | read   | reference |
| time          | longword      | uns longword | read   | reference |

#### Returns

##### **status\_return**

Specifies whether the routine completed successfully. GRAB\_POINTER returns the following status:

| Value            | Description                     |
|------------------|---------------------------------|
| XSC_GRAB_SUCCESS | Routine completed successfully. |

| Value                  | Description                                                                                                                                                                                                                              |
|------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| X\$C_ALREADY_GRABBED   | Pointer is actively grabbed by another client.                                                                                                                                                                                           |
| X\$C_GRAB_FROZEN       | Pointer is frozen by an active grab of another client.                                                                                                                                                                                   |
| X\$C_GRAB_INVALID_TIME | Time specified in <b>time</b> is earlier than the last pointer grab time, or later than the current server time. Otherwise, the last pointer grab time is set to the specified time. CurrentTime is replaced by the current server time. |
| X\$C_GRAB_NOT_VIEWABLE | Windows specified in <b>window_id</b> or <b>confine_id</b> are not currently viewable, or <b>confine_id</b> window lies completely outside the boundaries of the root window.                                                            |

## Arguments

### **event\_mask**

A bit mask that specifies the events. Table 2–5 lists the predefined values for the event mask.

**Table 2–5 Event Mask Description**

| Bit | Predefined Value         | Description                          |
|-----|--------------------------|--------------------------------------|
| 2   | X\$M_BUTTON_PRESS        | Pointer button down events wanted    |
| 3   | X\$M_BUTTON_RELEASE      | Pointer button up events wanted      |
| 4   | X\$M_ENTER_WINDOW        | Pointer window entry events wanted   |
| 5   | X\$M_LEAVE_WINDOW        | Pointer window leave events wanted   |
| 6   | X\$M_POINTER_MOTION      | Pointer motion events wanted         |
| 7   | X\$M_POINTER_MOTION_HINT | Pointer motion hints wanted          |
| 8   | X\$M_BUTTON1_MOTION      | Pointer motion while button 1 down   |
| 9   | X\$M_BUTTON2_MOTION      | Pointer motion while button 2 down   |
| 10  | X\$M_BUTTON3_MOTION      | Pointer motion while button 3 down   |
| 11  | X\$M_BUTTON4_MOTION      | Pointer motion while button 4 down   |
| 12  | X\$M_BUTTON5_MOTION      | Pointer motion while button 5 down   |
| 13  | X\$M_BUTTON_MOTION       | Pointer motion while any button down |
| 14  | X\$M_KEYMAP_STATE        | Any keyboard state change wanted     |

### **pointer\_mode**

The mode that the pointer events will use. The predefined values for **pointer\_mode** are as follows:

X\$C\_GRAB\_MODE\_SYNC  
X\$C\_GRAB\_MODE\_ASYNC

Other values specified in this argument are not valid.

## Xlib Routines

### GRAB POINTER

#### **keyboard\_mode**

The processing of keyboard events. The predefined values for **keyboard\_mode** are as follows:

X\$C\_GRAB\_MODE\_SYNC  
X\$C\_GRAB\_MODE\_ASYNC

Other values specified in this argument are not valid.

---

## GRAB SERVER

### OpenVMS Format

X\$GRAB\_SERVER  
(display)

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## HEIGHT MM OF SCREEN

### OpenVMS Format

height\_return = X\$HEIGHT\_MM\_OF\_SCREEN  
(screen\_id)

#### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| height_return | longword   | longword     | write  | value     |
| screen_id     | identifier | uns longword | read   | reference |

---

## HEIGHT OF SCREEN

### OpenVMS Format

height\_return = X\$HEIGHT\_OF\_SCREEN  
(screen\_id)

**Argument Information**

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| height_return | longword   | longword     | write  | value     |
| screen_id     | identifier | uns longword | read   | reference |

---

**ICONIFY WINDOW**

**OpenVMS Format**

status\_return = X\$ICONIFY\_WINDOW  
(display, window\_id, screen\_number)

**Argument Information**

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| status_return | longword     | uns longword | write  | reference |
| display       | identifier   | uns longword | read   | reference |
| window_id     | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |

**Returns**

**status\_return**  
Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

**IF EVENT**

**OpenVMS Format**

X\$IF\_EVENT  
(display, event\_return, predicate, arg)

## Xlib Routines IF EVENT

### Argument Information

| Argument     | Usage      | Data Type       | Access | Mechanism |
|--------------|------------|-----------------|--------|-----------|
| display      | identifier | uns longword    | read   | reference |
| event_return | record     | x\$event        | write  | reference |
| predicate    | procedure  | proc entry mask | read   | reference |
| arg          | longword   | uns longword    | read   | value     |

---

## IMAGE BYTE ORDER

### OpenVMS Format

order\_return = X\$IMAGE\_BYTE\_ORDER  
(display)

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| order_return | longword   | longword     | write  | value     |
| display      | identifier | uns longword | read   | reference |

### Returns

#### order\_return

The byte order for images for each scanline unit in XYFormat (bitmap) or for each pixel value in ZFormat. IMAGE BYTE ORDER returns one of the following constants:

| Value          | Description                 |
|----------------|-----------------------------|
| X\$C_LSB_FIRST | Least significant bit first |
| X\$C_MSB_FIRST | Most significant bit first  |

---

## INSERT MODIFIERMAP ENTRY

### OpenVMS Format

X\$INSERT\_MODIFIERMAP\_ENTRY  
(modifier\_keys, keycode\_entry, modifier, modifier\_keys\_return)



**Argument Information**

| <b>Argument</b>      | <b>Usage</b> | <b>Data Type</b>   | <b>Access</b> | <b>Mechanism</b> |
|----------------------|--------------|--------------------|---------------|------------------|
| modifier_keys        | record       | x\$modifier_keymap | read          | reference        |
| keycode_entry        | identifier   | uns longword       | read          | reference        |
| modifier             | longword     | uns longword       | read          | reference        |
| modifier_keys_return | record       | x\$modifier_keymap | write         | reference        |

**Arguments**

**modifier**

The modifier for which you add a key symbol. There are eight modifiers in the order (starting from zero) shift, lock, control, mod1, mod2, mod3, mod4, and mod5. You can pass the integer value or one of the following constants:

- X\$C\_SHIFT\_MAP\_INDEX
- X\$C\_LOCK\_MAP\_INDEX
- X\$C\_CONTROL\_MAP\_INDEX
- X\$C\_MOD1\_MAP\_INDEX
- X\$C\_MOD2\_MAP\_INDEX
- X\$C\_MOD3\_MAP\_INDEX
- X\$C\_MOD4\_MAP\_INDEX
- X\$C\_MOD5\_MAP\_INDEX

**modifier\_keys\_return**

INSERT MODIFIER MAP ENTRY returns the revised modifier key map data structure.

**INSTALL COLORMAP**

**OpenVMS Format**

X\$INSTALL\_COLORMAP  
(display, colormap\_id)

**Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| display         | identifier   | uns longword     | read          | reference        |
| colormap_id     | identifier   | uns longword     | read          | reference        |

---

## INTERN ATOM

### OpenVMS Format

atom\_id\_return = X\$INTERN\_ATOM  
(display, atom\_name, only\_if\_exists)

### Argument Information

| Argument       | Usage       | Data Type        | Access | Mechanism  |
|----------------|-------------|------------------|--------|------------|
| atom_id_return | identifier  | uns longword     | write  | value      |
| display        | identifier  | uns longword     | read   | reference  |
| atom_name      | char_string | character string | read   | descriptor |
| only_if_exists | Boolean     | longword         | read   | reference  |

---

## INTERSECT REGION

### OpenVMS Format

X\$INTERSECT\_REGION  
(src\_region1\_id, src\_region2\_id, dst\_region\_id\_return)

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| src_region1_id       | identifier | uns longword | read   | reference |
| src_region2_id       | identifier | uns longword | read   | reference |
| dst_region_id_return | identifier | uns longword | write  | reference |

---

## KEYCODE TO KEYSYM

### OpenVMS Format

keysym\_return = X\$KEYCODE\_TO\_KEYSYM  
(display, keycode, index)

**Argument Information**

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| keysym_return | identifier | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| keycode       | word       | uns word     | read   | reference |
| index         | longword   | longword     | read   | reference |

---

## KEYSYM TO KEYCODE

**OpenVMS Format**

```
keycode_return = X$KEYSYM_TO_KEYCODE
    (display, keysym_id)
```

**Argument Information**

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| keycode_return | word       | uns word     | write  | value     |
| display        | identifier | uns longword | read   | reference |
| keysym_id      | identifier | uns longword | read   | reference |

---

## KEYSYM TO STRING

**OpenVMS Format**

```
status_return = X$KEYSYM_TO_STRING
    (keysym_id, keysym_name_return)
```

**Argument Information**

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| status_return      | cond_value  | uns longword     | write  | value      |
| keysym_id          | identifier  | uns longword     | read   | reference  |
| keysym_name_return | char_string | character string | write  | descriptor |

## Xlib Routines

### KEYSYM TO STRING

#### Returns

**status\_return**

Specifies whether the routine completed successfully.

#### Arguments

**keysym\_name\_return**

The name of the key symbol string.

---

## KILL CLIENT

#### OpenVMS Format

X\$KILL\_CLIENT

(display, resource)

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| resource | longword   | longword     | read   | reference |

---

## LAST KNOWN REQUEST PROCESSED

#### OpenVMS Format

request\_return = X\$LAST\_KNOWN\_REQUEST\_PROCESSED

(display)

#### Argument Information

| Argument       | Usage        | Data Type    | Access | Mechanism |
|----------------|--------------|--------------|--------|-----------|
| request_return | uns longword | uns longword | write  | value     |
| display        | identifier   | uns longword | read   | reference |

---

## LIST DEPTHS

### OpenVMS Format

status\_return = X\$LIST\_DEPTH  
(display, screen\_number, count\_return, depths\_return)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| status_return | cond_value   | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |
| count_return  | longword     | uns longword | write  | reference |
| depths_return | address      | array        | write  | reference |

---

## LIST FONT

### OpenVMS Format

status\_return = X\$LIST\_FONT  
(display, pattern\_name, context, name, [len\_return])

### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism  |
|---------------|-------------|------------------|--------|------------|
| status_return | cond_value  | uns longword     | write  | value      |
| display       | identifier  | uns longword     | read   | reference  |
| pattern_name  | char_string | character string | read   | descriptor |
| context       | context     | uns longword     | modify | reference  |
| name          | char_string | character string | write  | descriptor |
| len_return    | word        | uns word         | write  | reference  |

---

## LIST FONTS

### OpenVMS Format

```
status_return = X$LIST_FONTS  
(display, pattern_name, match_limit, actual_count_return, names_return,  
 [len_return])
```

### Argument Information

| Argument            | Usage       | Data Type        | Access | Mechanism  |
|---------------------|-------------|------------------|--------|------------|
| status_return       | cond_value  | uns longword     | write  | value      |
| display             | identifier  | uns longword     | read   | reference  |
| pattern_name        | char_string | character string | read   | descriptor |
| match_limit         | longword    | longword         | read   | reference  |
| actual_count_return | longword    | longword         | write  | value      |
| names_return        | char_string | character string | write  | descriptor |
| len_return          | longword    | longword         | write  | reference  |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

| Value      | Description                     |
|------------|---------------------------------|
| SS\$NORMAL | Routine completed successfully. |

### Arguments

**names\_return**  
A character string containing all the returned font names, separated by commas.

**len\_return**  
Length of the returned string of font names. This argument is optional.

---

## LIST FONT WITH INFO

### OpenVMS Format

```
status_return = X$LIST_FONT_WITH_INFO  
(display, pattern_name, context, font_name_return, [len_return], font_struct_return)
```

### Argument Information

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| status_return      | cond_value  | uns longword     | write  | value      |
| display            | identifier  | uns longword     | read   | reference  |
| pattern_name       | char_string | character string | read   | descriptor |
| context            | context     | uns longword     | modify | reference  |
| font_name_return   | char_string | character string | write  | descriptor |
| len_return         | word        | uns word         | write  | reference  |
| font_struct_return | record      | x\$font_struct   | write  | reference  |

### Returns

#### status\_return

Specifies whether the routine completed successfully. LIST FONTS returns one of the following values:

| Value           | Description                                           |
|-----------------|-------------------------------------------------------|
| SS\$_NORMAL     | Routine completed successfully.                       |
| X\$_NOTFOUND    | No fonts matched pattern.                             |
| LIB\$_STRTRU    | Font names returned but truncated.                    |
| LIB\$_FATALERR  | LIB\$SCOPY_R_DX fatal error.                          |
| LIB\$_INSVIRMEM | Insufficient virtual memory; LIB\$GET_VM call failed. |
| LIB\$_INVSTRDES | Invalid string descriptor.                            |

### Arguments

#### font\_name\_return

The virtual address of a descriptor that points to a character string. The string containing the font name is returned by the routine and resides in space reserved by Xlib.

#### len\_return

The length of the returned string of font names. This argument is optional.

#### font\_struct\_return

The address of the font data structure associated with the font.

---

## LIST FONTS WITH INFO

### OpenVMS Format

status\_return = X\$LIST\_FONTS\_WITH\_INFO

(display, pattern\_name, maxnames, count\_return, font\_names\_return, [len\_return],  
[info\_return], [info\_size], [info\_buff\_return])

### Argument Information

| Argument          | Usage        | Data Type        | Access | Mechanism  |
|-------------------|--------------|------------------|--------|------------|
| status_return     | cond_value   | uns longword     | write  | value      |
| display           | identifier   | uns longword     | read   | reference  |
| pattern_name      | char_string  | character string | read   | descriptor |
| maxnames          | longword     | longword         | read   | reference  |
| count_return      | longword     | longword         | write  | reference  |
| font_names_return | char_string  | character string | write  | descriptor |
| len_return        | word         | uns word         | write  | reference  |
| info_return       | address      | uns longword     | read   | reference  |
| info_size         | longword     | longword         | read   | reference  |
| info_buff_return  | uns longword | uns longword     | write  | reference  |

### Returns

#### status\_return

Return value that specifies whether the routine completed successfully. LIST FONTS WITH INFO returns one of the following values:

| Value           | Description                                           |
|-----------------|-------------------------------------------------------|
| SS\$_NORMAL     | Routine completed successfully.                       |
| X\$_NOTFOUND    | No fonts matched pattern.                             |
| LIB\$_STRTRU    | Font names returned but truncated.                    |
| LIB\$_FATALERR  | LIB\$SCOPY_R_DX fatal error.                          |
| LIB\$_INSVIRMEM | Insufficient virtual memory, LIB\$GET_VM call failed. |
| LIB\$_INVSTRDES | Invalid string descriptor.                            |

### Arguments

#### len\_return

The length of the string of font names returned in **font\_names\_return**.

#### info\_return

The virtual address of a pointer to an array of font information data, returned by the routine and residing in space reserved by Xlib.



**info\_size**

The size of the buffer specified in **info\_buff\_return**.

**info\_buff\_return**

A pointer to a data buffer residing in space space you have reserved. Each entry is one font information element. The length of the buffer is specified by **info\_size**. The property data is returned by the routine.

## LIST HOSTS

### OpenVMS Format

```
status_return = X$LIST_HOSTS
    (display, num_hosts_return, state_return, [hosts_return], [hosts_size],
    [hosts_buff_return])
```

### Argument Information

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| status_return     | cond_value | uns longword | write  | value     |
| display           | identifier | uns longword | read   | reference |
| num_hosts_return  | longword   | longword     | write  | reference |
| state_return      | longword   | longword     | write  | reference |
| hosts_return      | address    | uns longword | write  | reference |
| hosts_size        | longword   | longword     | read   | reference |
| hosts_buff_return | array      | uns longword | write  | reference |

### Returns

**status\_return**

Specifies whether the routine completed successfully. Possible status values returned by the OpenVMS binding are as follows:

| Value          | Description                                                            |
|----------------|------------------------------------------------------------------------|
| X\$_ERRORREPLY | Error received from server.                                            |
| X\$_NOHOSTS    | No hosts available to make connections.                                |
| X\$_TRUNCATED  | User buffer specified in <b>time_buff_return</b> was not large enough. |
| SS\$_NORMAL    | Routine completed successfully.                                        |

### Arguments

**hosts\_return**

The virtual address of the hosts buffer, which contains the current access control list, is returned. This argument is optional. If you specify this argument, LIST HOSTS determines the size of the hosts buffer to create. If you specify **hosts\_return**, you do not need to specify **hosts\_size** and **hosts\_buff\_return**.

## Xlib Routines

### LIST HOSTS

#### **hosts\_size**

The size of the hosts buffer to which LIST HOSTS returns the list of hosts that can access a display. This argument is optional.

#### **hosts\_buff\_return**

A pointer to an array of addresses residing in space space you have reserved, where each element is the address of a host. The length of the array is specified by **num\_hosts\_return**. This argument is optional.

---

## LIST INSTALLED COLORMAPS

### OpenVMS Format

status\_return = X\$LIST\_INSTALLED\_COLORMAPS

(display, window\_id, num\_colormaps\_return, [colormaps\_return], [colormaps\_size], [colormaps\_buff\_return])

### Argument Information

| Argument              | Usage      | Data Type    | Access | Mechanism |
|-----------------------|------------|--------------|--------|-----------|
| status_return         | cond_value | uns longword | write  | value     |
| display               | identifier | uns longword | read   | reference |
| window_id             | identifier | uns longword | read   | reference |
| num_colormaps_return  | longword   | longword     | write  | reference |
| colormaps_return      | address    | uns longword | write  | reference |
| colormaps_size        | longword   | longword     | read   | reference |
| colormaps_buff_return | array      | uns longword | write  | reference |

### Returns

#### **status\_return**

Specifies whether the routine completed successfully. Possible status values returned by the OpenVMS binding are as follows:

| Value         | Description                                                            |
|---------------|------------------------------------------------------------------------|
| 0             | None.                                                                  |
| X\$_TRUNCATED | User buffer specified in <b>time_buff_return</b> was not large enough. |
| SS\$_NORMAL   | Routine completed successfully.                                        |

## Arguments

### **colormaps\_return**

The address of the buffer where the list of color map identifiers is returned. This argument is optional. If you specify this argument, LIST INSTALLED COLORMAPS determines the size of the buffer to create. If you specify **colormaps\_return**, you do not need to specify **colormaps\_size** and **colormaps\_buff\_return**.

### **colormaps\_size**

The size of the color map buffer.

### **colormaps\_buff\_return**

A pointer to a buffer residing in space you have reserved, where the list of color map identifiers is returned.

## LIST PIXMAP FORMATS

### OpenVMS Format

status\_return = X\$LIST\_PIXMAP\_FORMATS

(display, count\_return, [formats\_return], [formats\_size], [formats\_buff\_return])

### Argument Information

| Argument            | Usage      | Data Type    | Access | Mechanism |
|---------------------|------------|--------------|--------|-----------|
| status_return       | cond_value | uns longword | write  | value     |
| display             | identifier | uns longword | read   | reference |
| count_return        | longword   | uns longword | write  | reference |
| formats_return      | address    | array        | write  | reference |
| formats_size        | longword   | uns longword | read   | reference |
| formats_buff_return | array      | uns longword | write  | reference |

## Arguments

### **formats\_return**

Pointer to a data buffer returned by the routine and residing in space reserved by Xlib. Each entry is a pixmap format structure. This argument is optional.

### **formats\_size**

Size of the **formats\_buff\_return** buffer that receives the pixmap data

### **formats\_buff\_return**

Pointer to a data buffer residing in space you have reserved, where each entry is a pixmap format structure. This argument is optional.

---

## LIST PROPERTIES

### OpenVMS Format

```
status_return = X$LIST_PROPERTIES  
(display, window_id, num_prop_return, [properties_return], [properties_size],  
[properties_buff_return])
```

### Argument Information

| Argument               | Usage      | Data Type    | Access | Mechanism |
|------------------------|------------|--------------|--------|-----------|
| status_return          | cond_value | uns longword | write  | value     |
| display                | identifier | uns longword | read   | reference |
| window_id              | identifier | uns longword | read   | reference |
| num_prop_return        | longword   | longword     | write  | reference |
| properties_return      | address    | uns longword | write  | reference |
| properties_size        | longword   | longword     | read   | reference |
| properties_buff_return | array      | longword     | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. LIST PROPERTIES returns one of the following values:

| Value         | Description                                               |
|---------------|-----------------------------------------------------------|
| SS\$NORMAL    | Routine completed successfully.                           |
| X\$ERRORREPLY | Error received from server; window no longer exists.      |
| X\$TRUNCATED  | Results truncated; user-supplied buffer not large enough. |

### Arguments

#### properties\_return

The virtual address of a pointer to an array of property data, returned by the routine and residing in space reserved by Xlib.

#### properties\_size

The size of the **properties\_buff\_return** buffer that will receive the property list.

#### properties\_buff\_return

A pointer to a data buffer returned by the routine and residing in space you have reserved, where each entry is one property element. The length of the buffer is specified by **properties\_size**.

---

## LOAD FONT

### OpenVMS Format

```
font_id = X$LOAD_FONT
        (display, font_name)
```

### Argument Information

| Argument  | Usage       | Data Type        | Access | Mechanism  |
|-----------|-------------|------------------|--------|------------|
| font_id   | identifier  | uns longword     | write  | value      |
| display   | identifier  | uns longword     | read   | reference  |
| font_name | char_string | character string | read   | descriptor |

---

## LOAD QUERY FONT

### OpenVMS Format

```
status_return = X$LOAD_QUERY_FONT
              (display, font_name, font_struct_return)
```

### Argument Information

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| status_return      | cond_value  | uns longword     | write  | value      |
| display            | identifier  | uns longword     | read   | reference  |
| font_name          | char_string | character string | read   | descriptor |
| font_struct_return | record      | x\$font_struct   | write  | reference  |

### Returns

#### status\_return

Specifies whether the routine completed successfully. LOAD QUERY FONT returns one of the following values:

| Value          | Description                        |
|----------------|------------------------------------|
| SS\$_NORMAL    | Routine completed successfully.    |
| X\$_NOTFOUND   | No fonts matched pattern.          |
| LIB\$_STRTRU   | Font names returned but truncated. |
| LIB\$_FATALERR | LIB\$SCOPY_R_DX fatal error.       |

## Xlib Routines

### LOAD QUERY FONT

| Value           | Description                                           |
|-----------------|-------------------------------------------------------|
| LIB\$_INSVIRMEM | Insufficient virtual memory; LIB\$GET_VM call failed. |
| LIB\$_INVSTRDES | Invalid string descriptor.                            |

### Arguments

#### **font\_struct\_return**

The address of the font data structure associated with the font.

---

## LOOKUP COLOR

### OpenVMS Format

status\_return = X\$LOOKUP\_COLOR

(display, colormap\_id, color\_name, [screen\_def\_return], [exact\_def\_return])

### Argument Information

| Argument          | Usage       | Data Type        | Access | Mechanism  |
|-------------------|-------------|------------------|--------|------------|
| status_return     | cond_value  | uns longword     | write  | value      |
| display           | identifier  | uns longword     | read   | reference  |
| colormap_id       | identifier  | uns longword     | read   | reference  |
| color_name        | char_string | character string | read   | descriptor |
| screen_def_return | record      | x\$color         | write  | reference  |
| exact_def_return  | record      | x\$color         | write  | reference  |

### Returns

#### **status\_return**

Specifies whether the routine completed successfully. LOOKUP COLOR returns one of the following values:

| Value          | Description                            |
|----------------|----------------------------------------|
| SS\$_NORMAL    | Routine completed successfully.        |
| X\$_ERRORREPLY | Color name does not exist in database. |

### Arguments

#### **screen\_def\_return**

The color data structure where the red, green, and blue values of the color most closely supported by the screen hardware are returned. The screen is determined from the specified color map. This argument is optional.

**exact\_def\_return**

The color definition data structure where the red, green, and blue values of the exact color as defined in the color database are returned. The pixel, pad, and flags members are not used. This argument is optional.

## LOOKUP KEYSYM

### OpenVMS Format

```
keysym_id_return = X$LOOKUP_KEYSYM
    (key_event, index)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| keysym_id_return | identifier | uns longword | write  | value     |
| key_event        | record     | x\$key_event | read   | reference |
| index            | longword   | longword     | read   | reference |

## LOOKUP STRING

### OpenVMS Format

```
buflen_return = X$LOOKUP_STRING
    (key_event, buff_return, num_bytes, keysym_id_return, compose_status_return)
```

### Argument Information

| Argument              | Usage      | Data Type         | Access | Mechanism  |
|-----------------------|------------|-------------------|--------|------------|
| buflen_return         | longword   | longword          | write  | value      |
| key_event             | record     | x\$key_event      | read   | reference  |
| buff_return           | address    | longword          | write  | descriptor |
| num_bytes             | longword   | uns longword      | read   | reference  |
| keysym_id_return      | identifier | uns longword      | write  | reference  |
| compose_status_return | record     | x\$compose_status | write  | reference  |

## Xlib Routines

### LOWER WINDOW

---

## LOWER WINDOW

### OpenVMS Format

X\$LOWER\_WINDOW

(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## MAP RAISED

### OpenVMS Format

X\$MAP\_RAISED

(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## MAP SUBWINDOWS

### OpenVMS Format

X\$MAP\_SUBWINDOWS

(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |



---

## MAP WINDOW

### OpenVMS Format

X\$MAP\_WINDOW  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## MASK EVENT

### OpenVMS Format

X\$MASK\_EVENT  
(display, event\_mask, event\_return)

### Argument Information

| Argument     | Usage         | Data Type    | Access | Mechanism |
|--------------|---------------|--------------|--------|-----------|
| display      | identifier    | uns longword | read   | reference |
| event_mask   | mask_longword | longword     | read   | reference |
| event_return | record        | x\$event     | write  | reference |

---

## MATCH VISUAL INFO

### OpenVMS Format

status\_return = X\$MATCH\_VISUAL\_INFO  
(display, screen\_number, depth, class, vinfo\_return)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | longword   | longword     | write  | value     |
| display       | identifier | uns longword | read   | reference |

## Xlib Routines

### MATCH VISUAL INFO

| Argument      | Usage        | Data Type      | Access | Mechanism |
|---------------|--------------|----------------|--------|-----------|
| screen_number | uns longword | uns longword   | read   | reference |
| depth         | longword     | longword       | read   | value     |
| class         | longword     | longword       | read   | value     |
| vinfos_return | record       | x\$visual_info | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## MAX CMAPS OF SCREEN

### OpenVMS Format

```
colormaps_return = X$MAX_CMAPS_OF_SCREEN  
(screen_id)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| colormaps_return | longword   | longword     | write  | value     |
| screen_id        | identifier | uns longword | read   | reference |

---

## MAX REQUEST SIZE

### OpenVMS Format

```
size_return = X$MAX_REQUEST_SIZE  
(display)
```

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| size_return | longword   | longword     | write  | value     |
| display     | identifier | uns longword | read   | reference |

---

## MIN CMAPS OF SCREEN

### OpenVMS Format

```
colormaps_return = X$MIN_CMAPS_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| colormaps_return | longword   | longword     | write  | value     |
| screen_id        | identifier | uns longword | read   | reference |

---

## MOVE RESIZE WINDOW

### OpenVMS Format

```
X$MOVE_RESIZE_WINDOW
    (display, window_id, x_coord, y_coord, width, height)
```

### Argument Information

| Argument  | Usage        | Data Type    | Access | Mechanism |
|-----------|--------------|--------------|--------|-----------|
| display   | identifier   | uns longword | read   | reference |
| window_id | identifier   | uns longword | read   | reference |
| x_coord   | longword     | longword     | read   | reference |
| y_coord   | longword     | longword     | read   | reference |
| width     | uns longword | uns longword | read   | reference |
| height    | uns longword | uns longword | read   | reference |

---

## MOVE WINDOW

### OpenVMS Format

```
X$MOVE_WINDOW
    (display, window_id, x_coord, y_coord)
```

## Xlib Routines MOVE WINDOW

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |
| x_coord   | longword   | longword     | read   | reference |
| y_coord   | longword   | longword     | read   | reference |

---

## NEW MODIFIERMAP

### OpenVMS Format

```
status_return = X$NEW_MODIFIERMAP  
(max_keys_per_mod, mkeymap_return)
```

### Argument Information

| Argument         | Usage      | Data Type              | Access | Mechanism |
|------------------|------------|------------------------|--------|-----------|
| status_return    | cond_value | uns longword           | write  | value     |
| max_keys_per_mod | longword   | longword               | read   | reference |
| mkeymap_return   | record     | x\$modifier_<br>keymap | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. NEW MODIFIERMAP returns one of the following values:

| Value      | Description                     |
|------------|---------------------------------|
| SS\$NORMAL | Routine completed successfully. |
| 0          | Malloc failed.                  |

### Arguments

#### mkeymap\_return

The new modifier key map structure.

---

## **NEXT EVENT**

### **OpenVMS Format**

X\$NEXT\_EVENT  
(display, event\_return)

### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| display         | identifier   | uns longword     | read          | reference        |
| event_return    | record       | x\$event         | write         | reference        |

---

## **NEXT REQUEST**

### **OpenVMS Format**

request\_return = X\$NEXT\_REQUEST  
(display)

### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| request_return  | uns longword | uns longword     | write         | value            |
| display         | identifier   | uns longword     | read          | reference        |

---

## **NO OP**

### **OpenVMS Format**

X\$NO\_OP  
(display)

### **Argument Information**

| <b>Argument</b> | <b>Usage</b> | <b>Data Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|--------------|------------------|---------------|------------------|
| display         | identifier   | uns longword     | read          | reference        |

---

## OFFSET REGION

### OpenVMS Format

X\$OFFSET\_REGION  
(region\_id, x\_offset, y\_offset)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| region_id | identifier | uns longword | read   | reference |
| x_offset  | longword   | longword     | read   | reference |
| y_offset  | longword   | longword     | read   | reference |

---

## OPEN DISPLAY

### OpenVMS Format

display\_return = X\$OPEN\_DISPLAY  
([display\_name])

### Argument Information

| Argument       | Usage       | Data Type        | Access | Mechanism  |
|----------------|-------------|------------------|--------|------------|
| display_return | identifier  | uns longword     | write  | value      |
| display_name   | char_string | character string | read   | descriptor |

### Arguments

#### display\_name

The name of the hardware display to which you want to connect. The **display\_name** argument is a string that has the following format:

hostname::number.screen

The string elements are as follows:

- hostname** The name of the host machine to which the display is physically connected. Specify zero if the client program and the server are running in the same CPU.
- number** The number of the server on that host machine. A single CPU can have one or more servers, which are usually numbered starting with zero.

**screen**            The number of the screen on that server. An X server can control multiple screens on one display. The screen sets an internal variable that can be accessed by using the DefaultScreen macro or the DEFAULT SCREEN function.

The **display\_name** argument, which is optional, is the address of the character string descriptor that points to the string. If you omit the **display\_name** argument, the OPEN DISPLAY routine translates the logical name DECW\$DISPLAY to find the display name.

## PARSE COLOR

### OpenVMS Format

```
status_return = X$PARSE_COLOR
              (display, colormap_id, color_name, screen_def_return)
```

### Argument Information

| Argument          | Usage       | Data Type        | Access | Mechanism  |
|-------------------|-------------|------------------|--------|------------|
| status_return     | cond_value  | uns longword     | write  | value      |
| display           | identifier  | uns longword     | read   | reference  |
| colormap_id       | identifier  | uns longword     | read   | reference  |
| color_name        | char_string | character string | read   | descriptor |
| screen_def_return | record      | x\$color         | write  | reference  |

### Returns

#### status\_return

Specifies whether the routine completed successfully. PARSE COLOR returns one of the following values:

| Value         | Description                     |
|---------------|---------------------------------|
| SS\$NORMAL    | Routine completed successfully. |
| X\$ERRORREPLY | Error received from server.     |

### Arguments

#### color\_name

The name of the color. The string can be either a color name string or a numeric specification. If you use a text string, the name must be supported by the color database maintained by the server. See the SYSS\$MANAGER:DECW\$RGB.COM file for a list of all named colors. Case is not significant.

The **color\_name** argument is the address of a character string descriptor that points to the string.

---

## PARSE GEOMETRY

### OpenVMS Format

mask\_return = X\$PARSE\_GEOMETRY

(parse\_string, [x\_coord\_return], [y\_coord\_return], [width\_return], [height\_return])

### Argument Information

| Argument       | Usage         | Data Type        | Access | Mechanism  |
|----------------|---------------|------------------|--------|------------|
| mask_return    | mask_longword | uns longword     | write  | value      |
| parse_string   | char_string   | character string | read   | descriptor |
| x_coord_return | longword      | longword         | write  | reference  |
| y_coord_return | longword      | longword         | write  | reference  |
| width_return   | longword      | uns longword     | write  | reference  |
| height_return  | longword      | uns longword     | write  | reference  |

### Returns

#### mask\_return

A bit mask that specifies which of four values (width, height, x-offset, and y-offset) were actually found in the string, and whether the x and y values are negative. Each bit indicates whether the corresponding value was found in the parsed string. For each value found, the corresponding argument is updated; for each value not found, the argument is left unchanged.

Table 2–6 lists the predefined values and their descriptions for the mask.

**Table 2–6 Parse Mask Bits**

| Bit | OpenVMS              | Description                                |
|-----|----------------------|--------------------------------------------|
| 1   | XSM_NO_VALUE         | Reserved                                   |
| 2   | XSM_X_VALUE          | The x-coordinate of the origin of a window |
| 3   | XSM_Y_VALUE          | The y-coordinate of the origin of a window |
| 4   | XSM_WIDTH_VALUE      | The width of the window in pixels          |
| 5   | XSM_HEIGHT_VALUE     | The height of the window in pixels         |
| 6   | XSM_ALL_VALUES       | Indicates if all values are present        |
| 7   | XSM_X_NEGATIVE_VALUE | Indicates if the x-coordinate is negative  |
| 8   | XSM_Y_NEGATIVE_VALUE | Indicates if the y-coordinate is negative  |



## Arguments

### **parse\_string**

The name of the string that you want to parse. The **parse\_string** argument is the address of a character string descriptor that points to the string.

### **x\_coord\_return**

The x-coordinate to which to return the x-offset from the specified string. This coordinate is relative to the origin of the drawable. This argument is optional.

### **y\_coord\_return**

The y-coordinate to which to return the y-offset from the specified string. This coordinate is relative to the origin of the drawable. This argument is optional.

### **width\_return**

The width, in pixels, from the specified string. This argument is optional.

### **height\_return**

The height, in pixels, from the specified string. This argument is optional.

## PEEK EVENT

### OpenVMS Format

```
X$PEEK_EVENT
    (display, event_return)
```

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| display      | identifier | uns longword | read   | reference |
| event_return | record     | x\$event     | write  | reference |

## PEEK IF EVENT

### OpenVMS Format

```
X$PEEK_IF_EVENT
    (display, event_return, predicate, arg)
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## Xlib Routines

### PEEK IF EVENT

---

| Argument     | Usage     | Data Type       | Access | Mechanism |
|--------------|-----------|-----------------|--------|-----------|
| event_return | record    | x\$event        | write  | reference |
| predicate    | procedure | proc entry mask | read   | reference |
| arg          | longword  | uns longword    | read   | value     |

---

## PENDING

### OpenVMS Format

count\_return = X\$PENDING  
(display)

### Argument Information

---

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| count_return | longword   | longword     | write  | value     |
| display      | identifier | uns longword | read   | reference |

---

## PERM ALLOC

### OpenVMS Format

location\_return = X\$PERM\_ALLOC  
(size)

### Argument Information

---

| Argument        | Usage    | Data Type | Access | Mechanism |
|-----------------|----------|-----------|--------|-----------|
| location_return | longword | longword  | write  | value     |
| size            | longword | longword  | read   | reference |

---

## PLANES OF SCREEN

### OpenVMS Format

planes\_return = X\$PLANES\_OF\_SCREEN  
(screen\_id)

**Argument Information**

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| planes_return | longword   | longword     | write  | value     |
| screen_id     | identifier | uns longword | read   | reference |

**POINT IN REGION**

**OpenVMS Format**

```
answer_return = X$POINT_IN_REGION
    (region_id, x_coord, y_coord)
```

**Argument Information**

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| answer_return | longword   | longword     | write  | value     |
| region_id     | identifier | uns longword | read   | reference |
| x_coord       | longword   | longword     | read   | reference |
| y_coord       | longword   | longword     | read   | reference |

**POLYGON REGION**

**OpenVMS Format**

```
region_id_return = X$POLYGON_REGION
    (points, num_points, fill_rule)
```

**Argument Information**

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| region_id_return | identifier | uns longword | write  | value     |
| points           | array      | uns longword | read   | reference |
| num_points       | longword   | longword     | read   | reference |
| fill_rule        | longword   | uns longword | read   | reference |

## Xlib Routines

### POLYGON REGION

#### Returns

**region\_id\_return**

POLYGON REGION returns the region identifier when the region is created.

#### Arguments

**fill\_rule**

The fill rule that you want to set for the specified graphics context. The fill rule defines which pixels are inside (drawn) for paths given in FILL POLYGON requests.

The possible predefined values for **fill\_rule** are as follows:

XSC\_EVEN\_ODD\_RULE  
XSC\_WINDING\_RULE

---

## PROTOCOL REVISION

#### OpenVMS Format

revision\_return = X\$PROTOCOL\_REVISION  
(display)

#### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| revision_return | longword   | longword     | write  | value     |
| display         | identifier | uns longword | read   | reference |

---

## PROTOCOL VERSION

#### OpenVMS Format

version\_return = X\$PROTOCOL\_VERSION  
(display)

#### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| version_return | longword   | longword     | write  | value     |
| display        | identifier | uns longword | read   | reference |

---

## PUT BACK EVENT

### OpenVMS Format

X\$PUT\_BACK\_EVENT  
(display, event)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| event    | record     | x\$event     | read   | reference |

---

## PUT IMAGE

### OpenVMS Format

X\$PUT\_IMAGE  
(display, drawable\_id, gc\_id, ximage, src\_x\_coord, src\_y\_coord, dst\_x\_coord, dst\_y\_coord, width, height)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| drawable_id | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| ximage      | record     | x\$image     | read   | reference |
| src_x_coord | longword   | longword     | read   | reference |
| src_y_coord | longword   | longword     | read   | reference |
| dst_x_coord | longword   | longword     | read   | reference |
| dst_y_coord | longword   | longword     | read   | reference |
| width       | longword   | longword     | read   | reference |
| height      | longword   | longword     | read   | reference |

## Xlib Routines

### PUT PIXEL

---

## PUT PIXEL

### OpenVMS Format

X\$PUT\_PIXEL

(ximage, x\_coord, y\_coord, pixel)

### Argument Information

| Argument | Usage    | Data Type    | Access | Mechanism |
|----------|----------|--------------|--------|-----------|
| ximage   | record   | x\$image     | read   | reference |
| x_coord  | longword | longword     | read   | reference |
| y_coord  | longword | longword     | read   | reference |
| pixel    | longword | uns longword | read   | reference |

---

## Q LENGTH

### OpenVMS Format

length\_return = X\$Q\_LENGTH

(display)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| length_return | longword   | longword     | write  | value     |
| display       | identifier | uns longword | read   | reference |

---

## QUERY BEST CURSOR

### OpenVMS Format

X\$QUERY\_BEST\_CURSOR

(display, drawable\_id, width, height, width\_return, height\_return)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## Xlib Routines QUERY BEST CURSOR

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| drawable_id   | identifier | uns longword | read   | reference |
| width         | longword   | uns longword | read   | reference |
| height        | longword   | uns longword | read   | reference |
| width_return  | cond_value | uns longword | write  | reference |
| height_return | cond_value | uns longword | write  | reference |

### Returns

#### **status\_return**

Specifies whether the routine completed successfully.

| Value      | Description                     |
|------------|---------------------------------|
| SS\$NORMAL | Routine completed successfully. |

---

## QUERY BEST SIZE

### OpenVMS Format

status\_return = X\$QUERY\_BEST\_SIZE

(display, class, drawable\_id, width, height, width\_return, height\_return)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| status_return | cond_value   | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| class         | longword     | longword     | read   | reference |
| drawable_id   | identifier   | uns longword | read   | reference |
| width         | uns longword | uns longword | read   | reference |
| height        | uns longword | uns longword | read   | reference |
| width_return  | uns longword | uns longword | write  | reference |
| height_return | uns longword | uns longword | write  | reference |

### Returns

#### **status\_return**

Specifies whether the routine completed successfully. QUERY BEST SIZE returns one of the following values:

| Value         | Description                 |
|---------------|-----------------------------|
| X\$ERRORREPLY | Error received from server. |

## Xlib Routines

### QUERY BEST SIZE

| Value      | Description                     |
|------------|---------------------------------|
| SS\$NORMAL | Routine completed successfully. |

### Arguments

#### **class**

The item for which to determine the best size. The predefined values for **class** are as follows:

XSC\_TILE\_SHAPE  
XSC\_STIPPLE\_SHAPE  
XSC\_CURSOR\_SHAPE

Other values specified in this argument are not valid.

---

## QUERY BEST STIPPLE

### OpenVMS Format

```
status_return = X$QUERY_BEST_STIPPLE  
(display, drawable_id, width, height, width_return, height_return)
```

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| status_return | cond_value   | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| drawable_id   | identifier   | uns longword | read   | reference |
| width         | uns longword | uns longword | read   | reference |
| height        | uns longword | uns longword | read   | reference |
| width_return  | uns longword | uns longword | write  | reference |
| height_return | uns longword | uns longword | write  | reference |

### Returns

#### **status\_return**

Specifies whether the routine completed successfully. QUERY BEST STIPPLE returns one of the following values.

| Value         | Description                     |
|---------------|---------------------------------|
| X\$ERRORREPLY | Error received from server.     |
| SS\$NORMAL    | Routine completed successfully. |



---

## QUERY BEST TILE

### OpenVMS Format

status\_return = X\$QUERY\_BEST\_TILE  
(display, drawable\_id, width, height, width\_return, height\_return)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| status_return | cond_value   | uns longword | write  | value     |
| display       | identifier   | uns longword | read   | reference |
| drawable_id   | identifier   | uns longword | read   | reference |
| width         | uns longword | uns longword | read   | reference |
| height        | uns longword | uns longword | read   | reference |
| width_return  | uns longword | uns longword | write  | reference |
| height_return | uns longword | uns longword | read   | reference |

### Returns

**status\_return**  
Specifies whether the routine completed successfully. QUERY BEST TILE returns one of the following values:

| Value         | Description                     |
|---------------|---------------------------------|
| X\$ERRORREPLY | Error received from server.     |
| SS\$NORMAL    | Routine completed successfully. |

---

## QUERY COLOR

### OpenVMS Format

X\$QUERY\_COLOR  
(display, colormap\_id, screen\_def\_return)

### Argument Information

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| display           | identifier | uns longword | read   | reference |
| colormap_id       | identifier | uns longword | read   | reference |
| screen_def_return | record     | x\$color     | write  | reference |

---

## QUERY COLORS

### OpenVMS

X\$QUERY\_COLORS

(display, colormap\_id, screen\_defs\_return, num\_colors)

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| display            | identifier | uns longword | read   | reference |
| colormap_id        | identifier | uns longword | read   | reference |
| screen_defs_return | record     | x\$color     | write  | reference |
| num_colors         | longword   | longword     | read   | reference |

---

## QUERY FONT

### OpenVMS Format

status\_return = X\$QUERY\_FONT

(display, font\_id, font\_struct\_return)

### Argument Information

| Argument           | Usage      | Data Type      | Access | Mechanism |
|--------------------|------------|----------------|--------|-----------|
| status_return      | cond_value | uns longword   | write  | value     |
| display            | identifier | uns longword   | read   | reference |
| font_id            | identifier | uns longword   | read   | reference |
| font_struct_return | record     | x\$font_struct | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. QUERY FONT returns one of the following values:

| Value          | Description                        |
|----------------|------------------------------------|
| SS\$_NORMAL    | Routine completed successfully.    |
| X\$_NOTFOUND   | No fonts matched pattern.          |
| LIB\$_STRTRU   | Font names returned but truncated. |
| LIB\$_FATALERR | LIB\$SCOPY_R_DX fatal error.       |

| Value           | Description                                           |
|-----------------|-------------------------------------------------------|
| LIB\$_INSVIRMEM | Insufficient virtual memory; LIB\$GET_VM call failed. |
| LIB\$_INVSTRDES | Invalid string descriptor.                            |

## Arguments

### **font\_struct\_return**

The address of the font data structure associated with the font. To obtain character structure information from the font data structure, use the GET CHAR STRUCT routine after calling LOAD QUERY FONT.

---

## QUERY KEYMAP

### OpenVMS Format

X\$QUERY\_KEYMAP  
(display, keys\_return)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| keys_return | array      | byte         | write  | reference |

---

## QUERY POINTER

### OpenVMS Format

result\_return = X\$QUERY\_POINTER  
(display, window\_id, root\_id\_return, [child\_id\_return], [root\_x\_coord\_return], [root\_y\_coord\_return], [win\_x\_coord\_return], [win\_y\_coord\_return], [state\_mask\_return])

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| result_return  | Boolean    | longword     | write  | value     |
| display        | identifier | uns longword | read   | reference |
| window_id      | identifier | uns longword | read   | reference |
| root_id_return | identifier | uns longword | write  | reference |

## Xlib Routines

### QUERY POINTER

| Argument            | Usage        | Data Type    | Access | Mechanism |
|---------------------|--------------|--------------|--------|-----------|
| child_id_return     | identifier   | uns longword | write  | reference |
| root_x_coord_return | longword     | longword     | write  | reference |
| root_y_coord_return | longword     | longword     | write  | reference |
| win_x_coord_return  | longword     | longword     | write  | reference |
| win_y_coord_return  | longword     | longword     | write  | reference |
| state_mask_return   | uns longword | uns longword | write  | reference |

---

## QUERY TEXT EXTENTS

### OpenVMS Format

X\$QUERY\_TEXT\_EXTENTS

(display, font\_id, string, [direction\_return], [ascent\_return], [descent\_return],  
[overall\_return])

### Argument Information

| Argument         | Usage       | Data Type        | Access | Mechanism  |
|------------------|-------------|------------------|--------|------------|
| display          | identifier  | uns longword     | read   | reference  |
| font_id          | identifier  | uns longword     | read   | reference  |
| string           | char_string | character string | read   | descriptor |
| direction_return | cond_value  | uns longword     | write  | reference  |
| ascent_return    | cond_value  | uns longword     | write  | reference  |
| descent_return   | cond_value  | uns longword     | write  | reference  |
| overall_return   | cond_value  | x\$char_struct   | write  | reference  |

## ARGUMENTS

### direction\_return

The direction the string is painted on the screen. The **direction\_return** argument is the returned value of the direction element of the font data structure. This argument is optional.

### ascent\_return

The maximum ascent of the font used to draw the string. The **ascent\_return** argument is the returned value of the font ascent element of the font data structure. This argument is optional.

### descent\_return

The maximum descent of the font used to draw the string. The **descent\_return** argument is the returned value of the font descent element of the font data structure. This argument is optional.

**overall\_return**

The minimum left bearing, maximum right bearing, string width, maximum character ascent, and maximum character descent. Values are returned from a character structure. This argument is optional.

## QUERY TEXT EXTENTS 16

### OpenVMS Format

X\$QUERY\_TEXT\_EXTENTS\_16

(display, font\_id, string16, num\_chars, [direction\_return], [ascent\_return],  
[descent\_return], [overall\_return])

### Argument Information

| Argument         | Usage      | Data Type      | Access | Mechanism |
|------------------|------------|----------------|--------|-----------|
| display          | identifier | uns longword   | read   | reference |
| font_id          | identifier | uns longword   | read   | reference |
| string16         | array      | word           | read   | reference |
| num_chars        | word       | word           | read   | reference |
| direction_return | cond_value | uns longword   | write  | reference |
| ascent_return    | cond_value | uns longword   | write  | reference |
| descent_return   | cond_value | uns longword   | write  | reference |
| overall_return   | cond_value | x\$char_struct | read   | reference |

## ARGUMENTS

**direction\_return**

The direction in which the string is painted on the screen. The **direction\_return** argument is the returned value of the direction element of the font data structure. This argument is optional.

**ascent\_return**

Maximum ascent of the font used to draw the string. The **ascent\_return** argument is the returned value of the font ascent element of the font data structure. This argument is optional.

**descent\_return**

Maximum descent of the font used to draw the string. The **descent\_return** argument is the returned value of the font descent element of the font data structure. This argument is optional.

**overall\_return**

The minimum left bearing, maximum right bearing, string width, maximum character ascent, and maximum character descent. Values are returned from a character structure. This argument is optional.

---

## QUERY TREE

### OpenVMS Format

status\_return = X\$QUERY\_TREE

(display, window\_id, [root\_id\_return], [parent\_id\_return], [children\_return],  
[num\_children\_return], [children\_size\_return], [children\_buff\_return])

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| status_return        | cond_value | uns longword | write  | value     |
| display              | identifier | uns longword | read   | reference |
| window_id            | identifier | uns longword | read   | reference |
| root_id_return       | identifier | uns longword | write  | reference |
| parent_id_return     | identifier | uns longword | write  | reference |
| children_return      | address    | uns longword | write  | reference |
| num_children_return  | longword   | longword     | write  | reference |
| children_size_return | longword   | longword     | write  | reference |
| children_buff_return | identifier | uns longword | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. QUERY TREE returns one of the following values:

| Value          | Description                                          |
|----------------|------------------------------------------------------|
| X\$_ERRORREPLY | Error received from server; window no longer exists. |
| SS\$_NORMAL    | Routine completed successfully.                      |
| Otherwise      | Failure for reason given.                            |

### Arguments

#### root\_id\_return

The identifier of the root window of the specified window. This argument is optional.

#### parent\_id\_return

The identifier of the parent window of the specified window. This argument is optional.

**children\_return**

The virtual address of a pointer to an array of child windows, returned by the routine and residing in space reserved by Xlib. Each element in the array is a child window of the specified window. The number of elements in the array is specified by **num\_children\_return**. This argument is optional.

**num\_children\_return**

The number of children associated with the specified window. This argument is optional.

**children\_size\_return**

The size of the buffer containing the child windows, specified in **children\_buff\_return**. This argument is optional.

**children\_buff\_return**

A pointer to a data buffer residing in space you have reserved, where each entry is one child window identifier. The size of the buffer is specified by **children\_size\_return**. This argument is optional.

## RAISE WINDOW

### OpenVMS Format

```
X$RAISE_WINDOW
    (display, window_id)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

## READ BITMAP FILE

### OpenVMS Format

```
status_return = X$READ_BITMAP_FILE
    (display, drawable_id, filename, [width_return], [height_return], [bitmap_id_return],
    [x_hot_coord_return], [y_hot_coord_return])
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |

## Xlib Routines

### READ BITMAP FILE

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| display            | identifier  | uns longword     | read   | reference  |
| drawable_id        | identifier  | uns longword     | read   | reference  |
| filename           | char_string | character string | read   | descriptor |
| width_return       | longword    | uns longword     | write  | reference  |
| height_return      | longword    | uns longword     | write  | reference  |
| bitmap_id_return   | identifier  | uns longword     | write  | reference  |
| x_hot_coord_return | longword    | longword         | write  | reference  |
| y_hot_coord_return | longword    | longword         | write  | reference  |

### Returns

#### status\_return

READ BITMAP FILE returns one of the following values to indicate the status:

- XSC\_BITMAP\_SUCCESS
- XSC\_BITMAP\_OPEN\_FAILED
- XSC\_BITMAP\_FILE\_INVALID
- XSC\_BITMAP\_NO\_MEMORY

### Arguments

#### filename

The file specification of the bitmap file. The format of the file is dependent on the operating system on the client side of the client-server connection. File names are parsed using RMS\$PARSE and logical names, search strings, and so on, are supported. The maximum length of a file specification is 255 bytes. Wildcards are not supported. The default file name is [ ]bitmap.dat. The **filename** argument is the address of a character string descriptor that points to the string.

#### width\_return

The width of the read-in bitmap file. The **width\_return** argument is optional.

#### height\_return

The height of the read-in bitmap file. The **height\_return** argument is optional.

#### bitmap\_id\_return

The bitmap identifier. The **bitmap\_id\_return** argument is optional.

#### x\_hot\_coord\_return

The x-coordinate of the hotspot, which is defined as the point in the cursor that corresponds to the x- and y-coordinates reported for the pointer, is returned. If **x\_hot\_coord\_return** and **y\_hot\_coord\_return** are not null, READ BITMAP FILE sets **x\_hot\_coord\_return** and **y\_hot\_coord\_return** to the value of the hotspot as defined in the file. If no hotspot is defined, READ BITMAP FILE sets **x\_hot\_coord\_return** and **y\_hot\_coord\_return** to (-1, -1). The **x\_hot\_coord\_return** argument is optional.

#### y\_hot\_coord\_return

The y-coordinate of the hotspot, which is defined as the point in the cursor that corresponds to the x- and y-coordinates reported for the pointer, is returned. If **x\_hot\_coord\_return** and **y\_hot\_coord\_return** are not null, READ BITMAP FILE sets **x\_hot\_coord\_return** and **y\_hot\_coord\_return** to the value of the



hotspot as defined in the file. If no hotspot is defined, READ BITMAP FILE sets **x\_hot\_coord\_return** and **y\_hot\_coord\_return** to (-1, -1). The **y\_hot\_coord\_return** argument is optional.

## REBIND KEYSYM

### OpenVMS Format

X\$REBIND\_KEYSYM

(display, keysym\_id, keysym\_names, mod\_count, lookup\_string, num\_bytes)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| display       | identifier | uns longword | read   | reference |
| keysym_id     | identifier | uns longword | read   | reference |
| keysym_names  | array      | uns longword | read   | reference |
| mod_count     | longword   | longword     | read   | reference |
| lookup_string | word       | uns word     | read   | reference |
| num_bytes     | word       | uns word     | read   | reference |

## RECOLOR CURSOR

### OpenVMS Format

X\$RECOLOR\_CURSOR

(display, cursor\_id, foreground\_color, background\_color)

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| display          | identifier | uns longword | read   | reference |
| cursor_id        | identifier | uns longword | read   | reference |
| foreground_color | record     | x\$color     | read   | reference |
| background_color | record     | x\$color     | read   | reference |

## Xlib Routines

### RECONFIGURE WM WINDOW

---

## RECONFIGURE WM WINDOW

### OpenVMS Format

```
status_return = X$RECONFIGURE_WM_WINDOW  
(display, window_id, screen_number, change_mask, values)
```

### Argument Information

| Argument      | Usage             | Data Type             | Access | Mechanism |
|---------------|-------------------|-----------------------|--------|-----------|
| status_return | longword          | uns longword          | write  | value     |
| display       | identifier        | uns longword          | read   | reference |
| window_id     | identifier        | uns longword          | read   | reference |
| screen_number | uns longword      | uns longword          | read   | reference |
| change_mask   | mask_<br>longword | uns longword          | read   | reference |
| values        | record            | x\$window_<br>changes | read   | reference |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## RECT IN REGION

### OpenVMS Format

```
answer_return = X$RECT_IN_REGION  
(region_id, x_coord, y_coord, width, height)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| answer_return | longword   | longword     | write  | value     |
| region_id     | identifier | uns longword | read   | reference |
| x_coord       | longword   | longword     | read   | reference |

| Argument | Usage    | Data Type    | Access | Mechanism |
|----------|----------|--------------|--------|-----------|
| y_coord  | longword | longword     | read   | reference |
| width    | longword | uns longword | read   | reference |
| height   | longword | uns longword | read   | reference |

## Returns

### answer\_return

RECT IN REGION returns one of the following values:

XSC\_RECTANGLE\_IN  
XSC\_RECTANGLE\_OUT  
XSC\_RECTANGLE\_PART

---

## REFRESH KEYBOARD MAPPING

### OpenVMS Format

X\$REFRESH\_KEYBOARD\_MAPPING  
(event\_map)

### Argument Information

| Argument  | Usage  | Data Type            | Access | Mechanism |
|-----------|--------|----------------------|--------|-----------|
| event_map | record | x\$mapping_<br>event | read   | reference |

---

## REMOVE FROM SAVE SET

### OpenVMS Format

X\$REMOVE\_FROM\_SAVE\_SET  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

## Xlib Routines REMOVE HOST

---

### REMOVE HOST

#### OpenVMS Format

X\$REMOVE\_HOST

(display, host)

#### Argument Information

| Argument | Usage      | Data Type       | Access | Mechanism |
|----------|------------|-----------------|--------|-----------|
| display  | identifier | uns longword    | read   | reference |
| host     | record     | x\$host_address | read   | reference |

---

### REMOVE HOSTS

#### OpenVMS Format

X\$REMOVE\_HOSTS

(display, hosts, num\_hosts)

#### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| hosts     | array      | uns longword | read   | reference |
| num_hosts | longword   | longword     | read   | reference |

---

### REPARENT WINDOW

#### OpenVMS Format

X\$REPARENT\_WINDOW

(display, window\_id, parent\_id, x\_coord, y\_coord)

#### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| parent_id | identifier | uns longword | read   | reference |
| x_coord   | longword   | longword     | read   | reference |
| y_coord   | longword   | longword     | read   | reference |

---

## RESET SCREEN SAVER

### OpenVMS Format

X\$RESET\_SCREEN\_SAVER  
(display)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## RESIZE WINDOW

### OpenVMS Format

X\$RESIZE\_WINDOW  
(display, window\_id, width, height)

### Argument Information

| Argument  | Usage        | Data Type    | Access | Mechanism |
|-----------|--------------|--------------|--------|-----------|
| display   | identifier   | uns longword | read   | reference |
| window_id | identifier   | uns longword | read   | reference |
| width     | uns longword | uns longword | read   | reference |
| height    | uns longword | uns longword | read   | reference |

## Xlib Routines

### RESOURCE MANAGER STRING

---

## RESOURCE MANAGER STRING

### OpenVMS Format

```
status_return = X$RESOURCE_MANAGER_STRING  
(display, rm_string_return, [length_return])
```

### Argument Information

| Argument         | Usage       | Data Type        | Access | Mechanism  |
|------------------|-------------|------------------|--------|------------|
| status_return    | cond_value  | uns longword     | write  | value      |
| display          | identifier  | uns longword     | read   | reference  |
| rm_string_return | char_string | character string | write  | descriptor |
| length_return    | word        | word             | write  | reference  |

### Arguments

#### **rm\_string\_return**

A pointer to the resource manager string for the specified display.

#### **length\_return**

The length of the resource manager string. This argument is optional.

---

## RESTACK WINDOWS

### OpenVMS Format

```
X$RESTACK_WINDOWS  
(display, window_ids, num_windows)
```

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| window_ids  | record     | uns longword | read   | reference |
| num_windows | longword   | longword     | read   | reference |

## RM COMBINE DATABASE

### OpenVMS Format

```
X$RM_COMBINE_DATABASE
    (src_database_id, dst_database_id, override )
```

### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| src_database_id | identifier | uns longword | read   | reference |
| dst_database_id | identifier | uns longword | modify | reference |
| override        | Boolean    | uns longword | read   | reference |

## RM COMBINE FILE DATABASE

### OpenVMS Format

```
X$RM_COMBINE_FILE_DATABASE
    (file_name, dst_database_id, override)
```

### Argument Information

| Argument        | Usage       | Data Type        | Access | Mechanism  |
|-----------------|-------------|------------------|--------|------------|
| file_name       | char_string | character string | read   | descriptor |
| dst_database_id | identifier  | uns longword     | modify | reference  |
| override        | Boolean     | uns longword     | read   | reference  |

## RM DESTROY DATABASE

### OpenVMS Format

```
status_return = X$RM_DESTROY_DATABASE
    (database_id)
```

### Argument Information

| Argument      | Usage    | Data Type    | Access | Mechanism |
|---------------|----------|--------------|--------|-----------|
| status_return | longword | uns longword | write  | value     |

## Xlib Routines

### RM DESTROY DATABASE

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| database_id | identifier | uns longword | read   | reference |

#### Returns

##### status\_return

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## RM ENUMERATE DATABASE

### OpenVMS

```
status_return = X$RM_ENUMERATE_DATABASE
```

```
(database_id, name_list_id, class_list_id, mode, proc, closure )
```

#### Argument Information

| Argument      | Usage      | Data Type       | Access | Mechanism |
|---------------|------------|-----------------|--------|-----------|
| status_return | Boolean    | longword        | write  | value     |
| database_id   | identifier | uns longword    | read   | reference |
| name_list     | array      | uns longword    | read   | reference |
| class_list    | array      | uns longword    | read   | reference |
| mode          | longword   | longword        | read   | reference |
| proc          | procedure  | proc entry mask | read   | reference |
| closure       | longword   | uns longword    | read   | value     |

---

## RM GET DATABASE

### OpenVMS Format

```
database_id_return = X$RM_GET_DATABASE
```

```
(display )
```



**Argument Information**

| Argument               | Usage      | Data Type    | Access | Mechanism |
|------------------------|------------|--------------|--------|-----------|
| database_id_<br>return | identifier | uns longword | write  | value     |
| display                | identifier | uns longword | read   | reference |

---

**RM GET FILE DATABASE**

**OpenVMS Format**

```
database_id_return = X$RM_GET_FILE_DATABASE
    (file_name)
```

**Argument Information**

| Argument               | Usage       | Data Type           | Access | Mechanism  |
|------------------------|-------------|---------------------|--------|------------|
| database_id_<br>return | identifier  | uns longword        | write  | value      |
| file_name              | char_string | character<br>string | read   | descriptor |

---

**RM GET RESOURCE**

**OpenVMS Format**

```
status_return = X$RM_GET_RESOURCE
    (database_id, name_list_string, class_list_string, repr_type_return,
    [repr_value_return], [buf_len], [val_buf_return], [len_return])
```

**Argument Information**

| Argument             | Usage       | Data Type           | Access | Mechanism  |
|----------------------|-------------|---------------------|--------|------------|
| status_return        | cond_value  | uns longword        | write  | value      |
| database_id          | identifier  | uns longword        | read   | reference  |
| name_list_<br>string | char_string | character<br>string | read   | descriptor |
| class_list_string    | char_string | character<br>string | read   | descriptor |

## Xlib Routines

### RM GET RESOURCE

| Argument          | Usage        | Data Type        | Access | Mechanism  |
|-------------------|--------------|------------------|--------|------------|
| repr_type_return  | char_string  | character string | write  | descriptor |
| repr_value_return | record       | x\$rm_value      | write  | reference  |
| buf_len           | longword     | longword         | read   | reference  |
| val_buf_return    | uns longword | uns longword     | write  | reference  |
| len_return        | longword     | longword         | write  | reference  |

### Returns

#### status\_return

Specifies whether the routine completed successfully. RM GET RESOURCE returns one of the following values:

| Value         | Description                                               |
|---------------|-----------------------------------------------------------|
| SS\$_NORMAL   | Routine completed successfully.                           |
| X\$_NOTFOUND  | Resource not found.                                       |
| X\$_TRUNCATED | Results truncated; user-supplied buffer not large enough. |

### Arguments

#### val\_buf\_return

The address of the buffer containing the returned value. This argument is optional.

#### len\_return

The length of the returned value contained in the return value buffer. This argument is optional.

---

## RM GET STRING DATABASE

### OpenVMS Format

```
database_id = X$RM_GET_STRING_DATABASE
    (data)
```

### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| database_id | identifier  | uns longword     | write  | value      |
| data        | char_string | character string | read   | descriptor |

---

## RM INITIALIZE

### OpenVMS Format

status\_return = X\$RM\_INITIALIZE

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

---

## RM LOCALE OF DATABASE

### OpenVMS Format

status\_return = X\$RM\_LOCALE\_OF\_DATABASE  
(database\_id, locale\_string )

### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism  |
|---------------|-------------|------------------|--------|------------|
| status_return | cond_value  | uns longword     | write  | value      |
| database_id   | identifier  | uns longword     | read   | reference  |
| locale_string | char_string | character string | read   | descriptor |

---

## RM MERGE DATABASES

### OpenVMS Format

X\$RM\_MERGE\_DATABASES  
(src\_database\_id, dst\_database\_id)

## Xlib Routines

### RM MERGE DATABASES

#### Argument Information

| Argument        | Usage    | Data Type    | Access | Mechanism |
|-----------------|----------|--------------|--------|-----------|
| src_database_id | longword | uns longword | read   | reference |
| dst_database_id | longword | uns longword | modify | reference |

---

## RM PARSE COMMAND

### OpenVMS Format

X\$RM\_PARSE\_COMMAND

(database\_id, options, num\_options, prefix\_name, argc, argv)

#### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| database_id | identifier  | uns longword     | read   | reference  |
| options     | any         | uns longword     | read   | reference  |
| num_options | longword    | longword         | read   | reference  |
| prefix_name | char_string | character string | read   | descriptor |
| argc        | longword    | longword         | modify | reference  |
| argv        | any         | byte             | modify | reference  |

---

## RM PERM STRING TO QUARK

### OpenVMS Format

repr\_id\_return = X\$RM\_PERM\_STRING\_TO\_QUARK

(repr\_name )

#### Argument Information

| Argument       | Usage       | Data Type        | Access | Mechanism  |
|----------------|-------------|------------------|--------|------------|
| repr_id_return | identifier  | uns longword     | write  | value      |
| repr_name      | char_string | character string | read   | descriptor |

## RM PUT FILE DATABASE

### OpenVMS Format

X\$RM\_PUT\_FILE\_DATABASE  
(database\_id, file\_name)

### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| database_id | identifier  | uns longword     | read   | reference  |
| file_name   | char_string | character string | read   | descriptor |

## RM PUT LINE RESOURCE

### OpenVMS Format

X\$RM\_PUT\_LINE\_RESOURCE  
(database\_id, resource\_line)

### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism  |
|---------------|-------------|------------------|--------|------------|
| database_id   | record      | uns longword     | modify | reference  |
| resource_line | char_string | character string | read   | descriptor |

## RM PUT RESOURCE

### OpenVMS Format

X\$RM\_PUT\_RESOURCE  
(database\_id, specifier\_name, type\_name, [resource\_value], [buf\_len], [val\_buf])

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism  |
|-------------|------------|--------------|--------|------------|
| database_id | identifier | uns longword | modify | descriptor |

## Xlib Routines

### RM PUT RESOURCE

| Argument       | Usage        | Data Type        | Access | Mechanism  |
|----------------|--------------|------------------|--------|------------|
| specifier_name | char_string  | character string | read   | descriptor |
| type_name      | char_string  | character string | read   | descriptor |
| resource_value | record       | x\$rm_value      | read   | reference  |
| buf_len        | longword     | longword         | read   | reference  |
| val_buf        | uns longword | uns longword     | read   | reference  |

#### Arguments

**buf\_len**

Length of the value buffer. This argument is optional.

**val\_buf**

Address of the value buffer. This argument is optional.

---

## RM PUT STRING RESOURCE

#### OpenVMS Format

X\$RM\_PUT\_STRING\_RESOURCE

(database\_id, resource\_name, value\_name)

#### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism  |
|---------------|-------------|------------------|--------|------------|
| database_id   | identifier  | uns longword     | modify | reference  |
| resource_name | char_string | character string | read   | descriptor |
| value_name    | char_string | character string | read   | descriptor |

---

## RM Q GET RESOURCE

#### OpenVMS Format

status\_return = X\$RM\_Q\_GET\_RESOURCE

(database\_id, name\_list\_id, class\_list\_id, repr\_type\_id\_return,  
[repr\_value\_id\_return], [buf\_len], [val\_buf\_return], [len\_return])

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| status_return        | cond_value | uns longword | write  | value     |
| database_id          | identifier | uns longword | read   | reference |
| name_list_id         | identifier | uns longword | read   | reference |
| class_list_id        | identifier | uns longword | read   | reference |
| repr_type_id_return  | identifier | uns longword | write  | reference |
| repr_value_id_return | record     | x\$rm_value  | write  | reference |
| buf_len              | longword   | longword     | read   | reference |
| val_buf_return       | record     | byte         | write  | reference |
| len_return           | longword   | longword     | write  | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. RM Q GET RESOURCE returns one of the following values:

| Value         | Description                                               |
|---------------|-----------------------------------------------------------|
| SS\$_NORMAL   | Routine completed successfully.                           |
| X\$_NOTFOUND  | Resource not found.                                       |
| X\$_TRUNCATED | Results truncated; user-supplied buffer not large enough. |

### Arguments

#### buf\_len

The length of the buffer in which the value is returned. This argument is optional.

#### val\_buf\_return

The address of the buffer containing the returned value. This argument is optional.

#### len\_return

The length of the returned value contained in the return value buffer. This argument is optional.

## Xlib Routines

### RM Q GET SEARCH LIST

---

## RM Q GET SEARCH LIST

### OpenVMS Format

X\$RM\_Q\_GET\_SEARCH\_LIST

(database\_id, name\_list\_id, class\_list\_id, search\_list\_id\_return, list\_len)

### Argument Information

| Argument              | Usage        | Data Type    | Access | Mechanism |
|-----------------------|--------------|--------------|--------|-----------|
| database_id           | identifier   | uns longword | read   | reference |
| name_list_id          | identifier   | uns longword | read   | reference |
| class_list_id         | identifier   | uns longword | read   | reference |
| search_list_id_return | Boolean      | uns longword | write  | reference |
| list_len              | uns longword | uns longword | read   | reference |

---

## RM Q GET SEARCH RESOURCE

### OpenVMS Format

X\$RM\_Q\_GET\_SEARCH\_RESOURCE

(search\_list\_id, name\_id, class\_id, repr\_type\_id\_return, [repr\_value\_return], [buf\_len], [val\_buf\_return], [ret\_len\_return])

### Argument Information

| Argument            | Usage      | Data Type    | Access | Mechanism  |
|---------------------|------------|--------------|--------|------------|
| search_list_id      | identifier | uns longword | read   | reference  |
| name_id             | identifier | uns longword | read   | reference  |
| class_id            | identifier | uns longword | read   | reference  |
| repr_type_id_return | Boolean    | uns longword | write  | reference  |
| repr_value_return   | record     | xrm\$value   | write  | reference  |
| buf_len             | longword   | longword     | read   | reference  |
| val_buf_return      | record     | byte         | write  | descriptor |
| ret_len_return      | longword   | longword     | write  | reference  |

### Arguments

#### buf\_len

The length of the following buffer. This argument is optional.



**val\_buf\_return**

The returned buffer containing the value in the database. This argument is optional.

**ret\_len\_return**

The length of the data written to the buffer. This argument is optional.

---

## RM Q PUT RESOURCE

### OpenVMS Format

X\$RM\_Q\_PUT\_RESOURCE

(database\_id, binding\_list\_id, repr\_list\_id, repr\_type\_id, [repr\_value], [val\_len], [val\_buf])

### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| database_id     | identifier | uns longword | modify | reference |
| binding_list_id | identifier | uns longword | read   | reference |
| repr_list_id    | identifier | uns longword | read   | reference |
| repr_type_id    | identifier | uns longword | read   | reference |
| repr_value      | record     | x\$rm_value  | read   | reference |
| val_len         | longword   | longword     | read   | reference |
| val_buf         | any        | uns longword | read   | reference |

### Arguments

**repr\_value**

The descriptor for the resource entry. This argument is optional.

**val\_len**

Length of the value buffer. This argument is optional.

**val\_buf**

Address of the value buffer. This argument is optional.

---

## RM Q PUT STRING RESOURCE

### OpenVMS Format

X\$RM\_Q\_PUT\_STRING\_RESOURCE

(database\_id, binding\_list\_id, repr\_list\_id, value\_name)

## Xlib Routines

### RM Q PUT STRING RESOURCE

#### Argument Information

| Argument        | Usage       | Data Type        | Access | Mechanism  |
|-----------------|-------------|------------------|--------|------------|
| database_id     | identifier  | uns longword     | modify | reference  |
| binding_list_id | identifier  | uns longword     | read   | reference  |
| repr_list_id    | identifier  | uns longword     | read   | reference  |
| value_name      | char_string | character string | read   | descriptor |

---

## RM QUARK TO STRING

### OpenVMS Format

X\$RM\_QUARK\_TO\_STRING  
(repr\_id, repr\_name)

#### Argument Information

| Argument  | Usage       | Data Type        | Access | Mechanism  |
|-----------|-------------|------------------|--------|------------|
| repr_id   | identifier  | uns longword     | read   | reference  |
| repr_name | char_string | character string | write  | descriptor |

---

## RM SET DATABASE

### OpenVMS Format

X\$RM\_SET\_DATABASE  
(display, database\_id )

#### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| database_id | identifier | uns longword | read   | reference |

---

## RM STRING TO BIND QUARK LIST

### OpenVMS Format

```
X$RM_STRING_TO_BIND_QUARK_LIST  
(value_name, binding_list_id_return, repr_list_id_return)
```

### Argument Information

| Argument               | Usage       | Data Type        | Access | Mechanism  |
|------------------------|-------------|------------------|--------|------------|
| value_name             | char_string | character string | read   | descriptor |
| binding_list_id_return | identifier  | uns longword     | write  | reference  |
| repr_list_id_return    | array       | uns longword     | write  | reference  |

---

## RM STRING TO QUARK

### OpenVMS Format

```
repr_id_return = X$RM_STRING_TO_QUARK  
(repr_name)
```

### Argument Information

| Argument       | Usage       | Data Type        | Access | Mechanism  |
|----------------|-------------|------------------|--------|------------|
| repr_id_return | identifier  | uns longword     | write  | value      |
| repr_name      | char_string | character string | read   | descriptor |

---

## RM STRING TO QUARK LIST

### OpenVMS Format

```
X$RM_STRING_TO_QUARK_LIST  
(repr_name, repr_list_id_return)
```

## Xlib Routines

### RM STRING TO QUARK LIST

#### Argument Information

| Argument            | Usage       | Data Type        | Access | Mechanism  |
|---------------------|-------------|------------------|--------|------------|
| repr_name           | char_string | character string | read   | descriptor |
| repr_list_id_return | array       | uns longword     | write  | reference  |

---

## RM UNIQUE QUARK

### OpenVMS Format

repr\_id\_return = X\$RM\_UNIQUE\_QUARK

#### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| repr_id_return | identifier | uns longword | write  | value     |

---

## ROOT WINDOW

### OpenVMS Format

root\_window\_id\_return = X\$ROOT\_WINDOW

(display, screen\_number)

#### Argument Information

| Argument              | Usage        | Data Type    | Access | Mechanism |
|-----------------------|--------------|--------------|--------|-----------|
| root_window_id_return | identifier   | uns longword | write  | value     |
| display               | identifier   | uns longword | read   | reference |
| screen_number         | uns longword | uns longword | read   | reference |

## ROOT WINDOW OF SCREEN

### OpenVMS Format

```

window_id_return = X$ROOT_WINDOW_OF_SCREEN
    (screen_id)
    
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| window_id_return | identifier | uns longword | write  | value     |
| screen_id        | identifier | uns longword | read   | reference |

## ROTATE BUFFERS

### OpenVMS Format

```

X$ROTATE_BUFFERS
    (display, rotate)
    
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| rotate   | longword   | longword     | read   | reference |

## ROTATE WINDOW PROPERTIES

### OpenVMS Format

```

X$ROTATE_WINDOW_PROPERTIES
    (display, window_id, properties, num_prop, num_positions)
    
```

### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| window_id  | identifier | uns longword | read   | reference |
| properties | array      | uns longword | read   | reference |

## Xlib Routines

### ROTATE WINDOW PROPERTIES

| Argument      | Usage    | Data Type | Access | Mechanism |
|---------------|----------|-----------|--------|-----------|
| num_prop      | longword | longword  | read   | reference |
| num_positions | longword | longword  | read   | reference |

---

## SAVE CONTEXT

### OpenVMS Format

```
status_return = X$SAVE_CONTEXT  
(display, window_id, context_id, window_data, len)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| window_id     | identifier | uns longword | read   | reference |
| context_id    | identifier | uns longword | read   | reference |
| window_data   | longword   | uns longword | read   | reference |
| len           | longword   | longword     | read   | reference |

### Arguments

**len**  
Length of the data associated with the specified window and context type.

---

## SCREEN COUNT

### OpenVMS Format

```
count_return = X$SCREEN_COUNT  
(display)
```

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| count_return | longword   | longword     | write  | value     |
| display      | identifier | uns longword | read   | reference |

---

## SCREEN NUMBER OF SCREEN

### OpenVMS Format

identifier = X\$SCREEN\_NUMBER\_OF\_SCREEN  
(screen\_number)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| identifier    | identifier   | uns longword | write  | value     |
| screen_number | uns longword | uns longword | read   | reference |

---

## SCREEN OF DISPLAY

### OpenVMS Format

X\$SCREEN\_OF\_DISPLAY  
(display, screen\_number, screen\_return)

### Argument Information

| Argument      | Usage        | Data Type    | Access | Mechanism |
|---------------|--------------|--------------|--------|-----------|
| display       | identifier   | uns longword | read   | reference |
| screen_number | uns longword | uns longword | read   | reference |
| screen_return | record       | x\$screen    | write  | reference |

### Arguments

**screen\_return**  
The identifier of the screen associated with the specified display.

---

## SCREEN RESOURCE STRING

### OpenVMS Format

X\$SCREEN\_RESOURCE\_STRING  
(screen, res\_string)

## Xlib Routines

### SCREEN RESOURCE STRING

#### Argument Information

| Argument   | Usage       | Data Type        | Access | Mechanism  |
|------------|-------------|------------------|--------|------------|
| screen     | identifier  | uns longword     | read   | reference  |
| res_string | char_string | character string | write  | descriptor |

The res\_string descriptor will be filled in to point to memory allocated by Xlib; the caller is responsible for freeing this memory by calling X\$FREE.

---

## SELECT ASYNC EVENT

#### OpenVMS Format

X\$SELECT\_ASYNC\_EVENT  
(display, window\_id, event\_type, ast\_routine, ast\_userarg)

#### Argument Information

| Argument    | Usage      | Data Type       | Access | Mechanism |
|-------------|------------|-----------------|--------|-----------|
| display     | identifier | uns longword    | read   | reference |
| window_id   | identifier | uns longword    | read   | reference |
| event_type  | longword   | uns longword    | read   | reference |
| ast_routine | procedure  | proc entry mask | read   | reference |
| ast_userarg | longword   | uns longword    | read   | reference |

---

## SELECT ASYNC INPUT

#### OpenVMS Format

X\$SELECT\_ASYNC\_INPUT  
(display, window\_id, event\_mask, ast\_routine, ast\_userarg)

#### Argument Information

| Argument    | Usage      | Data Type       | Access | Mechanism |
|-------------|------------|-----------------|--------|-----------|
| display     | identifier | uns longword    | read   | reference |
| window_id   | identifier | uns longword    | read   | reference |
| event_mask  | longword   | uns longword    | read   | reference |
| ast_routine | procedure  | proc entry mask | read   | reference |
| ast_userarg | longword   | uns longword    | read   | reference |



---

## SELECT INPUT

### OpenVMS Format

X\$SELECT\_INPUT  
(display, window\_id, event\_mask)

### Argument Information

| Argument   | Usage         | Data Type    | Access | Mechanism |
|------------|---------------|--------------|--------|-----------|
| display    | identifier    | uns longword | read   | reference |
| window_id  | identifier    | uns longword | read   | reference |
| event_mask | mask_longword | longword     | read   | reference |

---

## SEND EVENT

### OpenVMS Format

status\_return = X\$SEND\_EVENT  
(display, window\_id, propagate, event\_mask, event)

### Argument Information

| Argument      | Usage         | Data Type    | Access | Mechanism |
|---------------|---------------|--------------|--------|-----------|
| status_return | longword      | uns longword | write  | value     |
| display       | identifier    | uns longword | read   | reference |
| window_id     | identifier    | uns longword | read   | reference |
| propagate     | longword      | longword     | read   | reference |
| event_mask    | mask_longword | longword     | read   | reference |
| event         | record        | x\$event     | read   | reference |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## Xlib Routines

### SEND EVENT

#### Arguments

##### **window\_id**

The identifier of the window to which to send the event. This window is called the destination window. You can pass the window identifier or the constants Pointer Window or Input Focus, which are as follows:

XSC\_POINTER\_WINDOW  
XSC\_INPUT\_FOCUS

---

## SERVER VENDOR

#### OpenVMS Format

X\$SERVER\_VENDOR  
(display, vendor\_name\_return, [len\_return])

#### Argument Information

| Argument           | Usage       | Data Type        | Access | Mechanism  |
|--------------------|-------------|------------------|--------|------------|
| display            | identifier  | uns longword     | read   | reference  |
| vendor_name_return | char_string | character string | write  | descriptor |
| len_return         | word        | uns word         | write  | reference  |

#### Arguments

##### **vendor\_name\_return**

The name of the string that identifies the owner of the X server implementation. The **vendor\_name\_return** argument is the address of a character string descriptor that points to the string.

##### **len\_return**

The length of the returned string. This argument is optional.

---

## SET ACCESS CONTROL

#### OpenVMS Format

X\$SET\_ACCESS\_CONTROL  
(display, access\_mode)

**Argument Information**

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| access_mode | longword   | longword     | read   | reference |

**Arguments**

**access\_mode**

Specifies whether you want to change the access control mode. The predefined values for **access\_mode** are as follows:

X\$C\_ENABLE\_ACCESS  
X\$C\_DISABLE\_ACCESS

**SET AFTER FUNCTION**

**OpenVMS Format**

X\$SET\_AFTER\_FUNCTION  
(display, func\_addr, [prev\_func\_addr\_return])

**Argument Information**

| Argument              | Usage      | Data Type       | Access | Mechanism |
|-----------------------|------------|-----------------|--------|-----------|
| display               | identifier | uns longword    | read   | reference |
| func_addr             | procedure  | proc entry mask | read   | reference |
| prev_func_addr_return | procedure  | proc entry mask | write  | reference |

**Arguments**

**prev\_func\_addr\_return**

The previous synchronization routine is returned.

**SET ARC MODE**

**OpenVMS Format**

X\$SET\_ARC\_MODE  
(display, gc\_id, arc\_mode)

## Xlib Routines

### SET ARC MODE

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc_id    | identifier | uns longword | read   | reference |
| arc_mode | longword   | longword     | read   | reference |

#### Arguments

##### **arc\_mode**

Specifies how an arc will be filled in a subsequent graphics request. The predefined values for **arc\_mode** are as follows:

| Value             | Description                                                                    |
|-------------------|--------------------------------------------------------------------------------|
| XSC_ARC_CHORD     | Only the chord area of the arc is filled.                                      |
| XSC_ARC_PIE_SLICE | The triangular area defined by the arc center and the arc endpoints is filled. |

Other values specified in this argument are not valid.

---

## SET BACKGROUND

#### OpenVMS

X\$SET\_BACKGROUND

(display, gc\_id, background)

#### Argument Information

| Argument   | Usage        | Data Type    | Access | Mechanism |
|------------|--------------|--------------|--------|-----------|
| display    | identifier   | uns longword | read   | reference |
| gc_id      | identifier   | uns longword | read   | reference |
| background | uns longword | uns longword | read   | reference |

---

## SET CLASS HINT

#### OpenVMS Format

X\$SET\_CLASS\_HINT

(display, window\_id, class\_hints\_return)

**Argument Information**

| Argument           | Usage      | Data Type     | Access | Mechanism |
|--------------------|------------|---------------|--------|-----------|
| display            | identifier | uns longword  | read   | reference |
| window_id          | identifier | uns longword  | read   | reference |
| class_hints_return | record     | x\$class_hint | write  | reference |

## SET CLIP MASK

**OpenVMS Format**

X\$SET\_CLIP\_MASK  
(display, gc\_id, pixmap\_id)

**Argument Information**

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| gc_id     | identifier | uns longword | read   | reference |
| pixmap_id | identifier | uns longword | read   | reference |

## SET CLIP ORIGIN

**OpenVMS Format**

X\$SET\_CLIP\_ORIGIN  
(display, gc\_id, clip\_x\_coord, clip\_y\_coord)

**Argument Information**

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| display      | identifier | uns longword | read   | reference |
| gc_id        | identifier | uns longword | read   | reference |
| clip_x_coord | longword   | longword     | read   | reference |
| clip_y_coord | longword   | longword     | read   | reference |

## Xlib Routines

### SET CLIP RECTANGLES

---

## SET CLIP RECTANGLES

### OpenVMS Format

X\$SET\_CLIP\_RECTANGLES

(display, gc\_id, clip\_x\_coord, clip\_y\_coord, rectangles, num\_rectangles, ordering)

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| display        | identifier | uns longword | read   | reference |
| gc_id          | identifier | uns longword | read   | reference |
| clip_x_coord   | longword   | longword     | read   | reference |
| clip_y_coord   | longword   | longword     | read   | reference |
| rectangles     | record     | x\$rectangle | read   | reference |
| num_rectangles | longword   | longword     | read   | reference |
| ordering       | longword   | longword     | read   | reference |

### Arguments

#### ordering

Specifies the ordering relationship of the clip rectangles.

The predefined values for **ordering** are as follows:

XSC\_UN\_SORTED  
XSC\_Y\_SORTED  
XSC\_Y\_X\_SORTED  
XSC\_Y\_X\_BANDED

Other values specified in this argument are not valid.

---

## SET CLOSE DOWN MODE

### OpenVMS Format

X\$SET\_CLOSE\_DOWN\_MODE

(display, close\_mode)

### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| close_mode | longword   | longword     | read   | reference |

## Arguments

### **close\_mode**

The close-down mode for the client's resources. The predefined values for **close\_mode** are as follows:

| Value                | Description                                   |
|----------------------|-----------------------------------------------|
| XSC_DESTROY_ALL      | All client resources are freed.               |
| XSC_RETAIN_PERMANENT | All client resources are marked as permanent. |
| XSC_RETAIN_TEMPORARY | All client resources are marked as temporary. |

---

## SET COMMAND

### OpenVMS Format

X\$SET\_COMMAND

(display, window\_id, command, num\_args)

### Argument Information

| Argument  | Usage       | Data Type        | Access | Mechanism  |
|-----------|-------------|------------------|--------|------------|
| display   | identifier  | uns longword     | read   | reference  |
| window_id | identifier  | uns longword     | read   | reference  |
| command   | char_string | character string | read   | descriptor |
| num_args  | longword    | longword         | read   | reference  |

---

## SET DASHES

### OpenVMS Format

X\$SET\_DASHES

(display, gc\_id, dash\_offset, dash\_list, dash\_list\_len)

### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| gc_id       | identifier | uns longword | read   | reference |
| dash_offset | longword   | longword     | read   | reference |
| dash_list   | byte       | byte         | read   | reference |

## Xlib Routines

### SET DASHES

| Argument      | Usage    | Data Type | Access | Mechanism |
|---------------|----------|-----------|--------|-----------|
| dash_list_len | longword | longword  | read   | reference |

---

## SET ERROR HANDLER

### OpenVMS Format

```
prev_handler_return = X$SET_ERROR_HANDLER  
    (handler)
```

### Argument Information

| Argument                | Usage     | Data Type       | Access | Mechanism |
|-------------------------|-----------|-----------------|--------|-----------|
| prev_handler_<br>return | procedure | proc entry mask | write  | reference |
| handler                 | procedure | proc entry mask | read   | reference |

### Arguments

#### handler

A user-written routine that handles nonfatal errors. The **handler** argument is the address of the entry mask for the routine. The routine returns an integer value. SET ERROR HANDLER passes the following arguments to the routine:

| Argument    | Description                                                           |
|-------------|-----------------------------------------------------------------------|
| display     | Pointer to the display structure associated with the display.         |
| error_event | Pointer to the error event. The error must be of type X\$ERROR_EVENT. |

---

## SET FILL RULE

### OpenVMS Format

```
X$SET_FILL_RULE  
    (display, gc_id, fill_rule)
```

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc_id    | identifier | uns longword | read   | reference |



| Argument  | Usage    | Data Type | Access | Mechanism |
|-----------|----------|-----------|--------|-----------|
| fill_rule | longword | longword  | read   | reference |

## Arguments

### fill\_rule

The fill rule that you want to set for the specified graphics context. The fill rule specifies which pixels are considered to be inside of a polygon for a FILL POLYGON routine. Those inside the polygon are then displayed, creating the fill. There are two predefined values:

XSC\_EVEN\_ODD\_RULE  
XSC\_WINDING\_RULE

---

## SET FILL STYLE

### OpenVMS Format

X\$SET\_FILL\_STYLE  
(display, gc\_id, fill\_style)

### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| gc_id      | identifier | uns longword | read   | reference |
| fill_style | longword   | longword     | read   | reference |

## Arguments

### fill\_style

The fill style for the space within the bounding box of a line, the even dash of an On Off Dash or Double Dash line, for an odd dash of a Double Dash line, for text lines, and for all fill requests. The predefined values for **fill\_style** are as follows:

XSC\_FILL\_SOLID  
XSC\_FILL\_TILED  
XSC\_FILL\_OPAQUE\_STIPPLED  
XSC\_FILL\_STIPPLED

Other values specified in this argument are not valid.

---

## SET FONT

### OpenVMS Format

X\$SET\_FONT  
(display, gc\_id, font\_id)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc_id    | identifier | uns longword | read   | reference |
| font_id  | identifier | uns longword | read   | reference |

---

## SET FONT PATH

### OpenVMS Format

X\$SET\_FONT\_PATH  
(display, directory\_names)

### Argument Information

| Argument        | Usage       | Data Type        | Access | Mechanism  |
|-----------------|-------------|------------------|--------|------------|
| display         | identifier  | uns longword     | read   | reference  |
| directory_names | char_string | character string | read   | descriptor |

---

## SET FOREGROUND

### OpenVMS Format

X\$SET\_FOREGROUND  
(display, gc\_id, foreground)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc_id    | identifier | uns longword | read   | reference |

| Argument   | Usage        | Data Type    | Access | Mechanism |
|------------|--------------|--------------|--------|-----------|
| foreground | uns longword | uns longword | read   | reference |

## SET FUNCTION

### OpenVMS

X\$SET\_FUNCTION  
(display, gc\_id, function)

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc_id    | identifier | uns longword | read   | reference |
| function | longword   | longword     | read   | reference |

### Arguments

#### function

The new value for the function member of the graphics context. The function member describes how the new destination bits are to be computed from the source bits and the old destination bits. Table 2-7 lists the valid values for **function**. The default value is 3 for X\$C\_GX\_COPY.

**Table 2-7 Graphics Context Codes for Function Member**

| Hex Value | OpenVMS Function Name | Operation             |
|-----------|-----------------------|-----------------------|
| 0         | X\$C_GX_CLEAR         | 0                     |
| 1         | X\$C_GX_AND           | src AND dst           |
| 2         | X\$C_GX_AND_REVERSE   | src AND NOT dst       |
| 3         | X\$C_GX_COPY          | src                   |
| 4         | X\$C_GX_AND_INVERTED  | (NOT src) AND dst     |
| 5         | X\$C_GX_NOOP          | dst                   |
| 6         | X\$C_GX_XOR           | src XOR dst           |
| 7         | X\$C_GX_OR            | src OR dst            |
| 8         | X\$C_GX_NOR           | (NOT src) AND NOT dst |
| 9         | X\$C_GX_EQUIV         | (NOT src) XOR dst     |
| A         | X\$C_GX_INVERT        | NOT dst               |
| B         | X\$C_GX_OR_REVERSE    | src OR NOT dst        |

(continued on next page)

## Xlib Routines

### SET FUNCTION

Table 2–7 (Cont.) Graphics Context Codes for Function Member

| Hex Value | OpenVMS Function Name | Operation            |
|-----------|-----------------------|----------------------|
| C         | X\$C_GX_COPY_INVERTED | NOT src              |
| D         | X\$C_GX_OR_INVERTED   | (NOT src) OR dst     |
| E         | X\$C_GX_NAND          | (NOT src) OR NOT dst |
| F         | X\$C_GX_SET           | 1                    |

## SET GRAPHICS EXPOSURES

### OpenVMS Format

X\$SET\_GRAPHICS\_EXPOSURES  
(display, gc\_id, graphics\_exposures)

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| display            | identifier | uns longword | read   | reference |
| gc_id              | identifier | uns longword | read   | reference |
| graphics_exposures | Boolean    | uns longword | read   | reference |

## SET ICON NAME

### OpenVMS Format

X\$SET\_ICON\_NAME  
(display, window\_id, icon\_name)

### Argument Information

| Argument  | Usage       | Data Type        | Access | Mechanism  |
|-----------|-------------|------------------|--------|------------|
| display   | identifier  | uns longword     | read   | reference  |
| window_id | identifier  | uns longword     | read   | reference  |
| icon_name | char_string | character string | read   | descriptor |

## SET ICON SIZES

### OpenVMS Format

X\$SET\_ICON\_SIZES  
(display, window\_id, size\_list, count)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |
| size_list | array      | uns longword | read   | reference |
| count     | longword   | longword     | read   | reference |

## SET INPUT FOCUS

### OpenVMS Format

X\$SET\_INPUT\_FOCUS  
(display, focus\_id, revert\_to, time)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| focus_id  | identifier | uns longword | read   | reference |
| revert_to | longword   | longword     | read   | reference |
| time      | longword   | uns longword | read   | reference |

### Arguments

#### focus\_id

The window identifier of the window in which you want to set the input focus.

The identifier of the window was originally returned by CREATE SIMPLE WINDOW or CREATE WINDOW. When the window identifier is specified, that window becomes the keyboard's focus window. When keyboard events are normally reported to this window (or its inferiors), the events continue to be reported. Otherwise, the event is reported with respect to the focus window.

One of the following predefined values can be specified instead of the window identifier:

X\$C\_POINTER\_ROOT  
X\$C\_NONE

## Xlib Routines

### SET INPUT FOCUS

#### revert\_to

Where the input focus moves to when the focus window becomes unviewable.

One of the following predefined values can be specified:

```
X$C_REVERT_TO_PARENT  
X$C_REVERT_TO_POINTER_ROOT  
X$C_REVERT_TO_NONE
```

---

## SET IO ERROR HANDLER

### OpenVMS Format

```
prev_handler_return = X$SET_IO_ERROR_HANDLER  
    (handler)
```

#### Argument Information

| Argument            | Usage     | Data Type       | Access | Mechanism |
|---------------------|-----------|-----------------|--------|-----------|
| prev_handler_return | procedure | proc entry mask | write  | reference |
| handler             | procedure | proc entry mask | read   | reference |

---

## SET LINE ATTRIBUTES

### OpenVMS Format

```
X$SET_LINE_ATTRIBUTES  
    (display, gc_id, line_width, line_style, cap_style, join_style)
```

#### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| gc_id      | identifier | uns longword | read   | reference |
| line_width | longword   | longword     | read   | reference |
| line_style | longword   | longword     | read   | reference |
| cap_style  | longword   | longword     | read   | reference |
| join_style | longword   | longword     | read   | reference |

## Arguments

### **line\_style**

Specifies the line style member. Line style defines the pattern of a line. The predefined values for **line\_style** are as follows:

```
XSC_LINE_SOLID
XSC_LINE_DOUBLE_DASH
XSC_LINE_OFF_DASH
```

Other values specified in this argument are not valid.

### **cap\_style**

Specifies the cap style member, which defines how the endpoints of a path are drawn. The predefined values for **cap\_style** are as follows:

```
XSC_CAP_NOT_LAST
XSC_CAP_BUTT
XSC_CAP_ROUND
XSC_CAP_PROJECTING
```

Other values specified in this argument are not valid. The default value is XSC\_CAP\_BUTT.

### **join\_style**

The join style. The join style defines how corners are drawn for wide lines. The predefined values for **join\_style** are as follows:

```
XSC_JOIN_MITER
XSC_JOIN_ROUND
XSC_JOIN_BEVEL
```

Other values specified in this argument are not valid. The default value is XSC\_JOIN\_MITER.

---

## SET MODIFIER MAPPING

### OpenVMS Format

```
status_return = X$SET_MODIFIER_MAPPING
    (display, modifier_keys)
```

### Argument Information

| Argument      | Usage      | Data Type          | Access | Mechanism |
|---------------|------------|--------------------|--------|-----------|
| status_return | cond_value | uns longword       | write  | value     |
| display       | identifier | uns longword       | read   | reference |
| modifier_keys | record     | x\$modifier_keymap | read   | reference |

## Xlib Routines

### SET MODIFIER MAPPING

#### Returns

##### **status\_return**

A server can impose restrictions on how modifiers can be changed. If such a restriction is violated, SET MODIFIER MAPPING returns a status message. SET MODIFIER MAPPING returns the following values:

XSC\_MAPPING\_SUCCESS  
XSC\_MAPPING\_FAILED  
XSC\_MAPPING\_BUSY

---

## SET NORMAL HINTS

#### OpenVMS Format

X\$SET\_NORMAL\_HINTS  
(display, window\_id, hints)

#### Argument Information

| Argument  | Usage      | Data Type     | Access | Mechanism |
|-----------|------------|---------------|--------|-----------|
| display   | identifier | uns longword  | read   | reference |
| window_id | identifier | uns longword  | read   | reference |
| hints     | record     | x\$size_hints | read   | reference |

---

## SET PLANE MASK

#### OpenVMS Format

X\$SET\_PLANE\_MASK  
(display, gc\_id, plane\_mask)

#### Argument Information

| Argument   | Usage         | Data Type    | Access | Mechanism |
|------------|---------------|--------------|--------|-----------|
| display    | identifier    | uns longword | read   | reference |
| gc_id      | identifier    | uns longword | read   | reference |
| plane_mask | mask_longword | uns longword | read   | reference |



---

## SET POINTER MAPPING

### OpenVMS Format

```
status_return = X$SET_POINTER_MAPPING
    (display, pointer_map, num_maps)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | cond_value | uns longword | write  | value     |
| display       | identifier | uns longword | read   | reference |
| pointer_map   | array      | byte         | read   | reference |
| num_maps      | word       | uns word     | read   | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. SET POINTER MAPPING returns one of the following status messages:

```
X$C_MAPPING_SUCCESS
X$C_MAPPING_BUSY
X$C_MAPPING_FAILED
```

---

## SET REGION

### OpenVMS Format

```
X$SET_REGION
    (display, gc_id, region_id)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| gc_id     | identifier | uns longword | read   | reference |
| region_id | identifier | uns longword | read   | reference |

---

## SET RGB COLORMAPS

### OpenVMS Format

X\$SET\_RGB\_COLORMAPS

(display, window\_id, standard\_colormap, count, property\_id)

### Argument Information

| Argument          | Usage      | Data Type                | Access | Mechanism |
|-------------------|------------|--------------------------|--------|-----------|
| display           | identifier | uns longword             | read   | reference |
| window_id         | identifier | uns longword             | read   | reference |
| standard_colormap | record     | x\$standard_<br>colormap | read   | reference |
| count             | longword   | longword                 | read   | reference |
| property_id       | identifier | uns longword             | read   | reference |

---

## SET SCREEN SAVER

### OpenVMS Format

X\$SET\_SCREEN\_SAVER

(display, timeout, interval, prefer\_blanking, allow\_exposures)

### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| display         | identifier | uns longword | read   | reference |
| timeout         | longword   | longword     | read   | reference |
| interval        | longword   | longword     | read   | reference |
| prefer_blanking | longword   | longword     | read   | reference |
| allow_exposures | longword   | longword     | read   | reference |

### Arguments

#### **prefer\_blanking**

Specifies the mode for whether to blank the screen during a screen save operation. The predefined values for **prefer\_blanking** are as follows:

X\$C\_DONT\_PREFER\_BLANKING  
X\$C\_PREFER\_BLANKING  
X\$C\_DEFAULT\_BLANKING

Other values specified in this argument are not valid.

**allow\_exposures**

Specifies screen saver control values. The predefined values for **allow\_exposures** are as follows:

X\$C\_DONT\_ALLOW\_EXPOSURES  
X\$C\_ALLOW\_EXPOSURES  
X\$C\_DEFAULT\_EXPOSURES

Other values specified in this argument are not valid.

## SET SELECTION OWNER

### OpenVMS Format

X\$SET\_SELECTION\_OWNER  
(display, selection\_id, owner\_window\_id, time)

### Argument Information

| Argument        | Usage      | Data Type    | Access | Mechanism |
|-----------------|------------|--------------|--------|-----------|
| display         | identifier | uns longword | read   | reference |
| selection_id    | identifier | uns longword | read   | reference |
| owner_window_id | identifier | uns longword | read   | reference |
| time            | longword   | uns longword | read   | reference |

## SET SIZE HINTS

### OpenVMS Format

X\$SET\_SIZE\_HINTS  
(display, window\_id, hints\_return, property)

### Argument Information

| Argument     | Usage      | Data Type     | Access | Mechanism |
|--------------|------------|---------------|--------|-----------|
| display      | identifier | uns longword  | read   | reference |
| window_id    | identifier | uns longword  | read   | reference |
| hints_return | record     | x\$size_hints | write  | reference |
| property     | identifier | uns longword  | read   | reference |

## Xlib Routines

### SET STANDARD COLORMAP

---

## SET STANDARD COLORMAP

### OpenVMS Format

X\$SET\_STANDARD\_COLORMAP

(display, window\_id, standard\_colormap, property\_id)

### Argument Information

| Argument          | Usage      | Data Type            | Access | Mechanism |
|-------------------|------------|----------------------|--------|-----------|
| display           | identifier | uns longword         | read   | reference |
| window_id         | identifier | uns longword         | read   | reference |
| standard_colormap | record     | x\$standard_colormap | read   | reference |
| property_id       | identifier | uns longword         | read   | reference |

---

## SET STANDARD PROPERTIES

### OpenVMS Format

X\$SET\_STANDARD\_PROPERTIES

(display, window\_id, [window\_name], [icon\_name], [icon\_pixmap], [command], [num\_args], [hints])

### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| display     | identifier  | uns longword     | read   | reference  |
| window_id   | identifier  | uns longword     | read   | reference  |
| window_name | char_string | character string | read   | descriptor |
| icon_name   | char_string | character string | read   | descriptor |
| icon_pixmap | identifier  | uns longword     | read   | reference  |
| command     | char_string | character string | read   | descriptor |
| num_args    | longword    | longword         | read   | reference  |
| hints       | record      | x\$size_hints    | read   | reference  |

---

## SET STATE

### OpenVMS Format

X\$SET\_STATE

(display, gc\_id, foreground, background, func, plane\_mask)

### Argument Information

| Argument   | Usage         | Data Type    | Access | Mechanism |
|------------|---------------|--------------|--------|-----------|
| display    | identifier    | uns longword | read   | reference |
| gc_id      | identifier    | uns longword | read   | reference |
| foreground | longword      | longword     | read   | reference |
| background | longword      | longword     | read   | reference |
| func       | longword      | longword     | read   | reference |
| plane_mask | mask_longword | uns longword | read   | reference |

### Arguments

#### func

The new value for the function member of the graphics context.

The function member describes how the new destination bits are computed from the source bits and the old destination bits. Table 2–8 lists the valid values for **func**.

**Table 2–8 Graphics Context Codes for Function Member**

| Hex Value | OpenVMS Function Name | Operation             |
|-----------|-----------------------|-----------------------|
| 0         | XSC_GX_CLEAR          | 0                     |
| 1         | XSC_GX_AND            | src AND dst           |
| 2         | XSC_GX_AND_REVERSE    | src AND NOT dst       |
| 3         | XSC_GX_COPY           | src                   |
| 4         | XSC_GX_AND_INVERTED   | (NOT src) AND dst     |
| 5         | XSC_GX_NOOP           | dst                   |
| 6         | XSC_GX_XOR            | src XOR dst           |
| 7         | XSC_GX_OR             | src OR dst            |
| 8         | XSC_GX_NOR            | (NOT src) AND NOT dst |
| 9         | XSC_GX_EQUIV          | (NOT src) XOR dst     |
| A         | XSC_GX_INVERT         | NOT dst               |
| B         | XSC_GX_OR_REVERSE     | src OR NOT dst        |
| C         | XSC_GX_COPY_INVERTED  | NOT src               |

(continued on next page)

## Xlib Routines

### SET STATE

Table 2–8 (Cont.) Graphics Context Codes for Function Member

| Hex Value | OpenVMS Function Name | Operation            |
|-----------|-----------------------|----------------------|
| D         | X\$C_GX_OR_INVERTED   | (NOT src) OR dst     |
| E         | X\$C_GX_NAND          | (NOT src) OR NOT dst |
| F         | X\$C_GX_SET           | 1                    |

---

## SET STIPPLE

### OpenVMS Format

X\$SET\_STIPPLE  
(display, gc\_id, stipple\_id)

### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| gc_id      | identifier | uns longword | read   | reference |
| stipple_id | identifier | longword     | read   | reference |

---

## SET SUBWINDOW MODE

### OpenVMS Format

X\$SET\_SUBWINDOW\_MODE  
(display, gc\_id, subwindow\_mode)

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| display        | identifier | uns longword | read   | reference |
| gc_id          | identifier | uns longword | read   | reference |
| subwindow_mode | longword   | longword     | read   | reference |

### Arguments

#### **subwindow\_mode**

Specifies whether the source and destination windows are clipped by subwindows. The predefined values are as follows:

X\$C\_CLIP\_BY\_CHILDREN

X\$C\_INCLUDE\_INFERIORS

Other values specified in this argument are not valid.

## SET TEXT PROPERTY

### OpenVMS Format

X\$SET\_TEXT\_PROPERTY

(display, window\_id, text\_prop, property\_id)

### Argument Information

| Argument    | Usage      | Data Type       | Access | Mechanism |
|-------------|------------|-----------------|--------|-----------|
| display     | identifer  | uns longword    | read   | reference |
| window_id   | identifier | uns longword    | read   | reference |
| text_prop   | record     | \$text_property | read   | reference |
| property_id | identifier | uns longword    | read   | reference |

## SET TILE

### OpenVMS Format

X\$SET\_TILE

(display, gc\_id, tile\_id)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| gc_id    | identifier | uns longword | read   | reference |
| tile_id  | identifier | uns longword | read   | reference |

## SET TRANSIENT FOR HINT

### OpenVMS Format

X\$SET\_TRANSIENT\_FOR\_HINT

(display, window\_id, prop\_window\_id)

## Xlib Routines

### SET TRANSIENT FOR HINT

#### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| display        | identifier | uns longword | read   | reference |
| window_id      | identifier | uns longword | read   | reference |
| prop_window_id | identifier | uns longword | read   | reference |

---

## SET TS ORIGIN

### OpenVMS Format

X\$SET\_TS\_ORIGIN  
(display, gc\_id, ts\_x\_coord, ts\_y\_coord)

#### Argument Information

| Argument   | Usage      | Data Type    | Access | Mechanism |
|------------|------------|--------------|--------|-----------|
| display    | identifier | uns longword | read   | reference |
| gc_id      | identifier | uns longword | read   | reference |
| ts_x_coord | longword   | longword     | read   | reference |
| ts_y_coord | longword   | longword     | read   | reference |

---

## SET WINDOW BACKGROUND

### OpenVMS Format

X\$SET\_WINDOW\_BACKGROUND  
(display, window\_id, pixel)

#### Argument Information

| Argument  | Usage        | Data Type    | Access | Mechanism |
|-----------|--------------|--------------|--------|-----------|
| display   | identifier   | uns longword | read   | reference |
| window_id | identifier   | uns longword | read   | reference |
| pixel     | uns longword | uns longword | read   | reference |



---

## SET WINDOW BACKGROUND PIXMAP

### OpenVMS Format

X\$SET\_WINDOW\_BACKGROUND\_PIXMAP  
(display, window\_id, background\_pixmap\_id)

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| display              | identifier | uns longword | read   | reference |
| window_id            | identifier | uns longword | read   | reference |
| background_pixmap_id | identifier | uns longword | read   | reference |

---

## SET WINDOW BORDER

### OpenVMS Format

X\$SET\_WINDOW\_BORDER  
(display, window\_id, pixel)

### Argument Information

| Argument  | Usage        | Data Type    | Access | Mechanism |
|-----------|--------------|--------------|--------|-----------|
| display   | identifier   | uns longword | read   | reference |
| window_id | identifier   | uns longword | read   | reference |
| pixel     | uns longword | uns longword | read   | reference |

---

## SET WINDOW BORDER PIXMAP

### OpenVMS Format

X\$SET\_WINDOW\_BORDER\_PIXMAP  
(display, window\_id, border\_pixmap\_id)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## Xlib Routines

### SET WINDOW BORDER PIXMAP

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| window_id        | identifier | uns longword | read   | reference |
| border_pixmap_id | identifier | uns longword | read   | reference |

---

### SET WINDOW BORDER WIDTH

#### OpenVMS Format

X\$SET\_WINDOW\_BORDER\_WIDTH

(display, window\_id, width)

#### Argument Information

| Argument  | Usage        | Data Type    | Access | Mechanism |
|-----------|--------------|--------------|--------|-----------|
| display   | identifier   | uns longword | read   | reference |
| window_id | identifier   | uns longword | read   | reference |
| width     | uns longword | uns longword | read   | reference |

---

### SET WINDOW COLORMAP

#### OpenVMS Format

X\$SET\_WINDOW\_COLORMAP

(display, window\_id, colormap\_id)

#### Argument Information

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| window_id   | identifier | uns longword | read   | reference |
| colormap_id | identifier | uns longword | read   | reference |

---

## SET WM CLIENT MACHINE

### OpenVMS Format

X\$SET\_CLIENT\_MACHINE  
(display, window\_id, text\_prop)

### Argument Information

| Argument  | Usage      | Data Type        | Access | Mechanism |
|-----------|------------|------------------|--------|-----------|
| display   | identifier | uns longword     | read   | reference |
| window_id | identifier | uns longword     | read   | reference |
| text_prop | record     | x\$text_property | read   | reference |

---

## SET WM COLORMAP WINDOWS

### OpenVMS Format

status\_return = X\$SET\_WM\_COLORMAP\_WINDOWS  
(display, window\_id, colormap\_windows, count)

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| status_return    | longword   | uns longword | write  | reference |
| display          | identifier | uns longword | read   | reference |
| window_id        | identifier | uns longword | read   | reference |
| colormap_windows | address    | array        | read   | reference |
| count            | longword   | longword     | read   | reference |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## Xlib Routines

### SET WM HINTS

---

## SET WM HINTS

### OpenVMS Format

X\$SET\_WM\_HINTS  
(display, window\_id, wmhints)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |
| wmhints   | record     | x\$wm_hints  | read   | reference |

---

## SET WM ICON NAME

### OpenVMS Format

X\$SET\_WM\_ICON\_NAME  
(display, window\_id, text\_prop)

### Argument Information

| Argument  | Usage      | Data Type        | Access | Mechanism |
|-----------|------------|------------------|--------|-----------|
| display   | identifier | uns longword     | read   | reference |
| window_id | address    | array            | read   | reference |
| text_prop | record     | x\$text_property | read   | reference |

---

## SET WM NAME

### OpenVMS Format

X\$SET\_WM\_NAME  
(display, window\_id, text\_prop)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

| Argument  | Usage      | Data Type        | Access | Mechanism |
|-----------|------------|------------------|--------|-----------|
| window_id | identifier | uns longword     | read   | reference |
| text_prop | record     | x\$text_property | read   | reference |

---

## SET WM NORMAL HINTS

### OpenVMS Format

X\$SET\_WM\_NORMAL\_HINTS  
(display, window\_id, hints)

### Argument Information

| Argument  | Usage      | Data Type     | Access | Mechanism |
|-----------|------------|---------------|--------|-----------|
| display   | identifier | uns longword  | read   | reference |
| window_id | identifier | uns longword  | read   | reference |
| hints     | record     | x\$size_hints | read   | reference |

---

## SET WM PROPERTIES

### OpenVMS Format

X\$SET\_WM\_PROPERTIES  
(display, window\_id, window\_name, icon\_name, command, num\_args,  
normal\_hints, wm\_hints, class\_hints)

### Argument Information

| Argument     | Usage       | Data Type        | Access | Mechanism |
|--------------|-------------|------------------|--------|-----------|
| display      | identifier  | uns longword     | read   | reference |
| window_id    | identifier  | uns longword     | read   | reference |
| window_name  | char_string | character string | read   | reference |
| icon_name    | char_string | character string | read   | reference |
| command      | char_string | character string | read   | reference |
| num_args     | longword    | longword         | read   | reference |
| normal_hints | record      | x\$size_hints    | read   | reference |
| wm_hints     | record      | x\$wm_hints      | read   | reference |
| class_hints  | record      | x\$class_hint    | read   | reference |

## Xlib Routines

### SET WM PROTOCOLS

---

## SET WM PROTOCOLS

### OpenVMS Format

```
status_return = X$SET_WM_PROTOCOLS  
(display, window_id, protocols, count)
```

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | longword   | uns longword | write  | reference |
| display       | identifier | uns longword | read   | reference |
| window_id     | identifier | uns longword | read   | reference |
| protocols     | identifier | uns longword | read   | reference |
| count         | longword   | longword     | read   | reference |

### Returns

**status\_return**  
Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

---

## SET WM SIZE HINTS

### OpenVMS Format

```
X$SET_WM_SIZE_HINTS  
(display, window_id, hints, property_id)
```

### Argument Information

| Argument    | Usage      | Data Type     | Access | Mechanism |
|-------------|------------|---------------|--------|-----------|
| display     | identifier | uns longword  | read   | reference |
| window_id   | identifier | uns longword  | read   | reference |
| hints       | record     | x\$size_hints | read   | reference |
| property_id | identifer  | uns longword  | read   | reference |

---

## SET ZOOM HINTS

### OpenVMS Format

X\$SET\_ZOOM\_HINTS  
(display, window\_id, zhints\_return)

### Argument Information

| Argument      | Usage      | Data Type     | Access | Mechanism |
|---------------|------------|---------------|--------|-----------|
| display       | identifier | uns longword  | read   | reference |
| window_id     | identifier | uns longword  | read   | reference |
| zhints_return | record     | x\$size_hints | read   | reference |

---

## SHRINK REGION

### OpenVMS Format

X\$SHRINK\_REGION  
(region\_id, x\_offset, y\_offset)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| region_id | identifier | uns longword | read   | reference |
| x_offset  | longword   | longword     | read   | reference |
| y_offset  | longword   | longword     | read   | reference |

---

## STORE BUFFER

### OpenVMS Format

X\$STORE\_BUFFER  
(display, bytes, num\_bytes, buffer)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

## Xlib Routines

### STORE BUFFER

| Argument  | Usage    | Data Type | Access | Mechanism |
|-----------|----------|-----------|--------|-----------|
| bytes     | array    | byte      | read   | reference |
| num_bytes | longword | longword  | read   | reference |
| buffer    | longword | longword  | read   | reference |

---

## STORE BYTES

### OpenVMS Format

X\$STORE\_BYTES  
(display, bytes, num\_bytes)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| bytes     | array      | byte         | read   | reference |
| num_bytes | longword   | longword     | write  | reference |

---

## STORE COLOR

### OpenVMS Format

X\$STORE\_COLOR  
(display, colormap\_id, screen\_def\_return)

### Argument Information

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| display           | identifier | uns longword | read   | reference |
| colormap_id       | identifier | uns longword | read   | reference |
| screen_def_return | record     | x\$color     | modify | reference |

### Arguments

#### screen\_def\_return

The color definition data structure that specifies the desired red, green, and blue color values; the color index where the color definition should be stored; and the flags member that specifies whether to set the red, green, or blue entries in



the color map. To set the flags member, do a bitwise OR with these predefined members:

```
X$M_DO_RED
X$M_DO_GREEN
X$M_DO_BLUE
```

The color stored is the color most closely supported by the hardware. The color definition must be a read/write entry.

## STORE COLORS

### OpenVMS Format

```
X$STORE_COLORS
    (display, colormap_id, screen_def_returns, num_colors)
```

### Argument Information

| Argument           | Usage      | Data Type    | Access     | Mechanism |
|--------------------|------------|--------------|------------|-----------|
| display            | identifier | uns longword | read       | reference |
| colormap_id        | identifier | uns longword | read       | reference |
| screen_def_returns | array      | x\$color     | read/write | reference |
| num_colors         | longword   | longword     | read       | reference |

### Arguments

#### screen\_def\_returns

The color definition data structure that specifies the desired red, green, and blue color values; the color index where the color definition should be stored; and the flags member that specifies whether to set the red, green, or blue entries in the color map. To set the flags member, do a bitwise OR with these predefined members:

```
X$M_DO_RED
X$M_DO_GREEN
X$M_DO_BLUE
```

The color stored is the color most closely supported by the hardware. The color definition must be a read/write entry.

---

## STORE NAME

### OpenVMS Format

X\$STORE\_NAME

(display, window\_id, window\_name)

### Argument Information

| Argument    | Usage       | Data Type        | Access | Mechanism  |
|-------------|-------------|------------------|--------|------------|
| display     | identifier  | uns longword     | read   | reference  |
| window_id   | identifier  | uns longword     | read   | reference  |
| window_name | char_string | character string | read   | descriptor |

---

## STORE NAMED COLOR

### OpenVMS Format

X\$STORE\_NAMED\_COLOR

(display, colormap\_id, color\_name, pixel, flags)

### Argument Information

| Argument    | Usage        | Data Type        | Access | Mechanism  |
|-------------|--------------|------------------|--------|------------|
| display     | identifier   | uns longword     | read   | reference  |
| colormap_id | identifier   | uns longword     | read   | reference  |
| color_name  | char_string  | character string | read   | descriptor |
| pixel       | uns longword | uns longword     | read   | reference  |
| flags       | uns longword | uns longword     | read   | reference  |

### Arguments

#### flags

Specifies whether to write the red, green, or blue values. Do a bitwise OR operation with these predefined values:

X\$M\_DO\_RED  
X\$M\_DO\_GREEN  
X\$M\_DO\_BLUE

## STRING LIST TO TEXT PROPERTY

### OpenVMS Format

```
status_return = X$STRING_LIST_TO_TEXT_PROPERTY
    (argv, argc, text_prop_return)
```

### Argument Information

| Argument         | Usage       | Data Type        | Access | Mechanism |
|------------------|-------------|------------------|--------|-----------|
| status_return    | longword    | uns longword     | write  | value     |
| argv             | char_string | character string | read   | reference |
| argc             | longword    | longword         | read   | reference |
| text_prop_return | record      | x\$text_property | write  | reference |

### Returns

**status\_return**  
 Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## STRING TO KEYSYM

### OpenVMS Format

```
keysym_return = X$STRING_TO_KEYSYM
    (keysym_name)
```

### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism  |
|---------------|-------------|------------------|--------|------------|
| keysym_return | identifier  | uns longword     | write  | value      |
| keysym_name   | char_string | character string | read   | descriptor |

## Xlib Routines

### SUB IMAGE

---

## SUB IMAGE

### OpenVMS Format

X\$SUB\_IMAGE

(ximage, x\_coord, y\_coord, width, height, sub\_image\_return)

### Argument Information

| Argument         | Usage    | Data Type    | Access | Mechanism |
|------------------|----------|--------------|--------|-----------|
| ximage           | record   | x\$image     | read   | reference |
| x_coord          | longword | longword     | read   | reference |
| y_coord          | longword | longword     | read   | reference |
| width            | longword | uns longword | read   | reference |
| height           | longword | uns longword | read   | reference |
| sub_image_return | record   | x\$image     | write  | reference |

---

## SUBTRACT REGION

### OpenVMS Format

X\$SUBTRACT\_REGION

(src\_region1\_id, src\_region2\_id, dst\_region\_id\_return)

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| src_region1_id       | identifier | uns longword | read   | reference |
| src_region2_id       | identifier | uns longword | read   | reference |
| dst_region_id_return | identifier | uns longword | write  | reference |

---

## SYNC

### OpenVMS Format

X\$SYNC

(display, discard)

**Argument Information**

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| discard  | longword   | longword     | read   | reference |

---

**SYNCHRONIZE**

**OpenVMS Format**

state\_return = X\$SYNCHRONIZE  
(display, onoff, after\_function\_return)

**Argument Information**

| Argument              | Usage      | Data Type       | Access | Mechanism |
|-----------------------|------------|-----------------|--------|-----------|
| state_return          | longword   | longword        | write  | value     |
| display               | identifier | uns longword    | read   | reference |
| onoff                 | longword   | longword        | read   | reference |
| after_function_return | procedure  | proc entry mask | write  | reference |

---

**TEXT EXTENTS**

**OpenVMS Format**

X\$TEXT\_EXTENTS  
(font\_ptr, string, [direction\_return], [ascent\_return], [descent\_return],  
[overall\_return])

**Argument Information**

| Argument         | Usage       | Data Type        | Access | Mechanism  |
|------------------|-------------|------------------|--------|------------|
| font_ptr         | record      | x\$font_struct   | read   | reference  |
| string           | char_string | character string | read   | descriptor |
| direction_return | longword    | longword         | write  | reference  |
| ascent_return    | longword    | longword         | write  | reference  |
| descent_return   | longword    | longword         | write  | reference  |
| overall_return   | record      | x\$char_struct   | write  | reference  |

## Xlib Routines

### TEXT EXTENTS

#### ARGUMENTS

##### **direction\_return**

The direction the font is painted on the screen. The **direction\_return** argument is the returned value of the direction element of the font data structure. This argument is optional.

##### **ascent\_return**

The maximum ascent of the font that draws the string. The **ascent\_return** argument is the returned value of the font ascent element of the font data structure. This argument is optional.

##### **descent\_return**

Maximum descent of the font that draws the string. The **descent\_return** argument is the returned value of the font descent element of the font data structure. This argument is optional.

##### **overall\_return**

The minimum left bearing, maximum right bearing, string width, maximum character ascent, and maximum character descent. This argument is optional.

---

## TEXT EXTENTS 16

### OpenVMS Format

X\$TEXT\_EXTENTS\_16

(font\_ptr, string, num\_chars, [direction\_return], [ascent\_return], [descent\_return], [overall\_return])

#### Argument Information

| Argument         | Usage    | Data Type      | Access | Mechanism |
|------------------|----------|----------------|--------|-----------|
| font_ptr         | record   | x\$font_struct | read   | reference |
| string           | array    | word           | read   | reference |
| num_chars        | word     | uns word       | read   | reference |
| direction_return | longword | longword       | write  | reference |
| ascent_return    | longword | longword       | write  | reference |
| descent_return   | longword | longword       | write  | reference |
| overall_return   | record   | x\$char_struct | write  | reference |

#### ARGUMENTS

##### **direction\_return**

The direction the string is painted on the screen. The **direction\_return** argument is the returned value of the direction element of the font data structure. This argument is optional.

**ascent\_return**

Maximum ascent of the font used to draw the string. The **ascent\_return** argument is the returned value of the font ascent element of the font data structure. This argument is optional.

**descent\_return**

Maximum descent of the font used to draw the string. The **descent\_return** argument is the returned value of the font descent element of the font data structure. This argument is optional.

**overall\_return**

The minimum left bearing, maximum right bearing, string width, maximum character ascent, and maximum character descent of the string. This argument is optional.

## TEXT PROPERTY TO STRING LIST

### OpenVMS Format

```
status_return = X$TEXT_PROPERTY_TO_STRING_LIST
    (text_prop, list_return, count_return)
```

### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism |
|---------------|-------------|------------------|--------|-----------|
| status_return | longword    | uns longword     | write  | value     |
| text_prop     | record      | x\$text_property | read   | reference |
| list_return   | char_string | character string | write  | reference |
| count_return  | longword    | longword         | write  | reference |

### Returns

**status\_return**

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## Xlib Routines

### TEXT WIDTH

---

## TEXT WIDTH

### OpenVMS Format

```
width_return = X$TEXT_WIDTH  
(font_ptr, string)
```

### Argument Information

| Argument     | Usage       | Data Type        | Access | Mechanism  |
|--------------|-------------|------------------|--------|------------|
| width_return | longword    | uns longword     | write  | value      |
| font_ptr     | record      | x\$font_struct   | read   | reference  |
| string       | char_string | character string | read   | descriptor |

---

## TEXT WIDTH 16

### OpenVMS Format

```
width_return = X$TEXT_WIDTH_16  
(font_ptr, string16, count)
```

### Argument Information

| Argument     | Usage    | Data Type      | Access | Mechanism |
|--------------|----------|----------------|--------|-----------|
| width_return | longword | uns longword   | write  | value     |
| font_ptr     | record   | x\$font_struct | read   | reference |
| string16     | array    | word           | read   | reference |
| count        | word     | uns word       | read   | reference |

---

## TRANSLATE COORDINATES

### OpenVMS Format

```
same_screen_return = X$TRANSLATE_COORDINATES  
(display, src_window_id, dst_window_id, src_x_coord, src_y_coord,  
 [dst_x_coord_return], [dst_y_coord_return], [child_id_return])
```



### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| same_screen_return | Boolean    | longword     | write  | value     |
| display            | identifier | uns longword | read   | reference |
| src_window_id      | identifier | uns longword | read   | reference |
| dst_window_id      | identifier | uns longword | read   | reference |
| src_x_coord        | longword   | longword     | read   | reference |
| src_y_coord        | longword   | longword     | read   | reference |
| dst_x_coord_return | longword   | longword     | write  | reference |
| dst_y_coord_return | longword   | longword     | write  | reference |
| child_id_return    | identifier | uns longword | write  | reference |

### Arguments

#### **dst\_x\_coord\_return**

The x-coordinate of the destination window. TRANSLATE COORDINATES returns the source coordinates relative to the origin of the source window to the **dst\_x\_coord\_return** and **dst\_y\_coord\_return** arguments, which are relative to the destination window's origin. This argument is optional.

#### **dst\_y\_coord\_return**

The y-coordinate of the destination window. TRANSLATE COORDINATES returns the source coordinates relative to the origin of the source window to the **dst\_x\_coord\_return** and **dst\_y\_coord\_return** arguments, which are relative to the destination window's origin. This argument is optional.

#### **child\_id\_return**

If the destination coordinates are contained in a mapped child of the destination window, the identifier of that child window is returned in **child\_id\_return**. This argument is optional.

---

## UNDEFINE CURSOR

### OpenVMS Format

```
X$UNDEFINE_CURSOR
    (display, window_id)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## UNGRAB BUTTON

### OpenVMS Format

X\$UNGRAB\_BUTTON

(display, button, modifiers, window\_id)

### Argument Information

| Argument  | Usage         | Data Type    | Access | Mechanism |
|-----------|---------------|--------------|--------|-----------|
| display   | identifier    | uns longword | read   | reference |
| button    | longword      | longword     | read   | reference |
| modifiers | mask_longword | uns longword | read   | reference |
| window_id | identifier    | uns longword | read   | reference |

### Arguments

#### button

The button on the pointing device that is no longer grabbed. The possible values are as follows:

XSC\_BUTTON1  
XSC\_BUTTON2  
XSC\_BUTTON3  
XSC\_BUTTON4  
XSC\_BUTTON5  
XSC\_ANY\_BUTTON

The predefined value XSC\_ANY\_BUTTON or AnyButton can be specified to allow any pointer button to be released.

#### modifiers

A bit mask that specifies the set of key masks associated with the button grab. This mask is the inclusive OR of these key mask bits:

| Bit | Predefined Value |
|-----|------------------|
| 1   | XSM_SHIFT        |
| 2   | XSM_CAPS_LOCK    |
| 3   | XSM_CONTROL      |
| 4   | XSM_MOD1         |
| 5   | XSM_MOD2         |
| 6   | XSM_MOD3         |
| 7   | XSM_MOD4         |
| 8   | XSM_MOD5         |

Clients can also pass the X\$ANY\_MODIFIER constants, which is equivalent to issuing the ungrab request for all possible modifier combinations (including the combination of no modifiers).

---

## UNGRAB KEY

### OpenVMS Format

X\$UNGRAB\_KEY  
(display, keycode, modifiers, window\_id)

### Argument Information

| Argument  | Usage             | Data Type    | Access | Mechanism |
|-----------|-------------------|--------------|--------|-----------|
| display   | identifier        | uns longword | read   | reference |
| keycode   | longword          | longword     | read   | reference |
| modifiers | mask_<br>longword | uns longword | read   | reference |
| window_id | identifier        | uns longword | read   | reference |

### Arguments

#### modifiers

A bit mask that specifies the set of key masks associated with the button grab. This mask is the inclusive OR of these key mask bits:

| Bit | Predefined Value |
|-----|------------------|
| 1   | X\$M_SHIFT       |
| 2   | X\$M_CAPS_LOCK   |
| 3   | X\$M_CONTROL     |
| 4   | X\$M_MOD1        |
| 5   | X\$M_MOD2        |
| 6   | X\$M_MOD3        |
| 7   | X\$M_MOD4        |
| 8   | X\$M_MOD5        |

Clients can also pass the X\$\_ANY\_MODIFIER constants, which is equivalent to issuing the ungrab request for all possible modifier combinations (including the combination of no modifiers).

---

## UNGRAB KEYBOARD

### OpenVMS Format

X\$UNGRAB\_KEYBOARD  
(display, time)

## Xlib Routines

### UNGRAB\_KEYBOARD

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| time     | longword   | uns longword | read   | reference |

---

## UNGRAB\_POINTER

### OpenVMS Format

X\$UNGRAB\_POINTER

(display, time)

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| time     | longword   | uns longword | read   | reference |

---

## UNGRAB\_SERVER

### OpenVMS Format

X\$UNGRAB\_SERVER

(display)

#### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |

---

## UNINSTALL\_COLORMAP

### OpenVMS Format

X\$UNINSTALL\_COLORMAP

(display, colormap\_id)

**Argument Information**

| Argument    | Usage      | Data Type    | Access | Mechanism |
|-------------|------------|--------------|--------|-----------|
| display     | identifier | uns longword | read   | reference |
| colormap_id | identifier | uns longword | read   | reference |

**UNION RECT WITH REGION**

**OpenVMS Format**

X\$UNION\_RECT\_WITH\_REGION  
(rectangle\_struct, src\_region\_id, dst\_region\_id\_return)

**Argument Information**

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| rectangle_struct     | record     | x\$rectangle | read   | reference |
| src_region_id        | identifier | uns longword | read   | reference |
| dst_region_id_return | identifier | uns longword | write  | reference |

**UNION REGION**

**OpenVMS Format**

X\$UNION\_REGION  
(src\_region1\_id, src\_region2\_id, dst\_region\_id\_return)

**Argument Information**

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| src_region1_id       | identifier | uns longword | read   | reference |
| src_region2_id       | identifier | uns longword | read   | reference |
| dst_region_id_return | identifier | uns longword | write  | reference |

## Xlib Routines

### UNIQUE CONTEXT

---

## UNIQUE CONTEXT

### OpenVMS Format

context\_id\_return = X\$UNIQUE\_CONTEXT

### Argument Information

| Argument          | Usage      | Data Type    | Access | Mechanism |
|-------------------|------------|--------------|--------|-----------|
| context_id_return | identifier | uns longword | write  | reference |

---

## UNLOAD FONT

### Format

X\$UNLOAD\_FONT  
(display, font\_id)

### Argument Information

| Argument | Usage      | Data Type    | Access | Mechanism |
|----------|------------|--------------|--------|-----------|
| display  | identifier | uns longword | read   | reference |
| font_id  | identifier | uns longword | read   | reference |

---

## UNMAP SUBWINDOWS

### OpenVMS Format

X\$UNMAP\_SUBWINDOWS  
(display, window\_id)

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## UNMAP WINDOW

### OpenVMS Format

```
X$UNMAP_WINDOW
    (display, window_id)
```

### Argument Information

| Argument  | Usage      | Data Type    | Access | Mechanism |
|-----------|------------|--------------|--------|-----------|
| display   | identifier | uns longword | read   | reference |
| window_id | identifier | uns longword | read   | reference |

---

## VENDOR RELEASE

### OpenVMS Format

```
release_return = X$VENDOR_RELEASE
    (display)
```

### Argument Information

| Argument       | Usage      | Data Type    | Access | Mechanism |
|----------------|------------|--------------|--------|-----------|
| release_return | longword   | longword     | write  | value     |
| display        | identifier | uns longword | read   | reference |

---

## VISUAL ID FROM VISUAL

### OpenVMS Format

```
visual_id_return = X$VISUAL_ID_FROM_VISUAL
    (visual_struct)
```

### Argument Information

| Argument         | Usage      | Data Type    | Access | Mechanism |
|------------------|------------|--------------|--------|-----------|
| visual_id_return | identifier | uns longword | write  | value     |
| visual_struct    | record     | x\$visual    | read   | reference |

## Xlib Routines

### WARP POINTER

---

## WARP POINTER

### OpenVMS Format

X\$WARP\_POINTER

(display, src\_window\_id, dst\_window\_id, src\_x\_coord, src\_y\_coord, src\_width,  
src\_height, dst\_x\_coord, dst\_y\_coord)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| display       | identifier | uns longword | read   | reference |
| src_window_id | identifier | uns longword | read   | reference |
| dst_window_id | identifier | uns longword | read   | reference |
| src_x_coord   | longword   | longword     | read   | reference |
| src_y_coord   | longword   | longword     | read   | reference |
| src_width     | longword   | uns longword | read   | reference |
| src_height    | longword   | uns longword | read   | reference |
| dst_x_coord   | longword   | longword     | read   | reference |
| dst_y_coord   | longword   | longword     | read   | reference |

---

## WHITE PIXEL

### OpenVMS Format

color\_index\_return = X\$WHITE\_PIXEL

(display, screen\_id)

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| color_index_return | identifier | uns longword | write  | value     |
| display            | identifier | uns longword | read   | reference |
| screen_id          | identifier | uns longword | read   | reference |



## WHITE PIXEL OF SCREEN

### OpenVMS Format

```
color_index_return = X$WHITE_PIXEL_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument           | Usage      | Data Type    | Access | Mechanism |
|--------------------|------------|--------------|--------|-----------|
| color_index_return | identifier | uns longword | write  | value     |
| screen_id          | identifier | uns longword | read   | reference |

## WIDTH MM OF SCREEN

### OpenVMS Format

```
width_return = X$WIDTH_MM_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| width_return | longword   | longword     | write  | value     |
| screen_id    | identifier | uns longword | read   | reference |

## WIDTH OF SCREEN

### OpenVMS Format

```
width_return = X$WIDTH_OF_SCREEN
    (screen_id)
```

### Argument Information

| Argument     | Usage      | Data Type    | Access | Mechanism |
|--------------|------------|--------------|--------|-----------|
| width_return | longword   | longword     | write  | value     |
| screen_id    | identifier | uns longword | read   | reference |

## Xlib Routines

### WINDOW EVENT

---

## WINDOW EVENT

### OpenVMS Format

X\$WINDOW\_EVENT

(display, window\_id, event\_mask, event\_return)

### Argument Information

| Argument     | Usage         | Data Type    | Access | Mechanism |
|--------------|---------------|--------------|--------|-----------|
| display      | identifier    | uns longword | read   | reference |
| window_id    | identifier    | uns longword | read   | reference |
| event_mask   | mask_longword | uns longword | read   | reference |
| event_return | record        | x\$event     | write  | reference |

---

## WITHDRAW WINDOW

### OpenVMS Format

status\_return = X\$WITHDRAW WINDOW

(display, window\_id, screen\_number)

### Argument Information

| Argument      | Usage      | Data Type    | Access | Mechanism |
|---------------|------------|--------------|--------|-----------|
| status_return | longword   | uns longword | write  | value     |
| display       | identifier | uns longword | read   | value     |
| window_id     | identifier | uns longword | read   | value     |
| screen_number | longword   | longword     | read   | value     |

### Returns

#### status\_return

Specifies whether the routine completed successfully.

| Value    | Description                            |
|----------|----------------------------------------|
| Non-zero | Routine completed successfully.        |
| 0        | Routine did not complete successfully. |

## WM GEOMETRY

### OpenVMS Format

mask = X\$WM\_GEOMETRY

(display, screen\_number, user\_geometry, default\_geometry, border\_width, hints,  
x\_return, y\_return, width\_return, height\_return, gravity\_return)

### Argument Information

| Argument         | Usage             | Data Type        | Access | Mechanism |
|------------------|-------------------|------------------|--------|-----------|
| mask             | mask_<br>longword | uns longword     | write  | value     |
| display          | identifier        | uns longword     | read   | reference |
| screen_number    | longword          | longword         | read   | reference |
| user_geometry    | char_string       | character string | read   | reference |
| default_geometry | char_string       | character string | read   | reference |
| border_width     | longword          | uns longword     | read   | reference |
| hints            | record            | x\$size_hints    | read   | reference |
| x_return         | longword          | longword         | write  | reference |
| y_return         | longword          | longword         | write  | reference |
| width_return     | longword          | longword         | write  | reference |
| height_return    | longword          | longword         | write  | reference |
| gravity_return   | longword          | longword         | write  | reference |

## WRITE BITMAP FILE

### OpenVMS Format

status\_return = X\$WRITE\_BITMAP\_FILE

(display, filename, bitmap\_id, width, height, x\_hot\_coord, y\_hot\_coord)

### Argument Information

| Argument      | Usage       | Data Type        | Access | Mechanism  |
|---------------|-------------|------------------|--------|------------|
| status_return | cond_value  | uns longword     | write  | value      |
| display       | identifier  | uns longword     | read   | reference  |
| filename      | char_string | character string | read   | descriptor |
| bitmap_id     | identifier  | uns longword     | read   | reference  |
| width         | longword    | uns longword     | read   | reference  |

## Xlib Routines

### WRITE BITMAP FILE

| Argument    | Usage    | Data Type    | Access | Mechanism |
|-------------|----------|--------------|--------|-----------|
| height      | longword | uns longword | read   | reference |
| x_hot_coord | longword | longword     | read   | reference |
| y_hot_coord | longword | longword     | read   | reference |

### Returns

#### status\_return

Specifies whether the routine completed successfully. WRITE BITMAP FILE returns one of the following values to indicate the status:

XSC\_BITMAP\_SUCCESS  
XSC\_BITMAP\_OPEN\_FAILED  
XSC\_BITMAP\_NO\_MEMORY

### Arguments

#### filename

The name of the file in which WRITE BITMAP FILE writes the bitmap. The format of the file is dependent on the operating system on the client side of the client-server connection. VMS logical names, search strings, and so on, are supported. The maximum length of a file specification is 255 bytes. Wildcards are not supported. The default file name is [ ]bitmap.dat. The **filename** argument is the address of a character string descriptor that points to the string.

---

## XOR REGION

### OpenVMS Format

X\$XOR\_REGION

(src\_region1\_id, src\_region2\_id, dst\_region\_id\_return)

### Argument Information

| Argument             | Usage      | Data Type    | Access | Mechanism |
|----------------------|------------|--------------|--------|-----------|
| src_region1_id       | identifier | uns longword | read   | reference |
| src_region2_id       | identifier | uns longword | read   | reference |
| dst_region_id_return | identifier | uns longword | write  | reference |

---

## Xlib Data Structures and Error Codes

Predefined data structures are provided for complex data structures used by Xlib routines. The OpenVMS binding format presents the data structure in a generic OpenVMS language. The generic language provides the OpenVMS binding names for the data structure and individual fields within the data structure. It also provides the basic OpenVMS data type for each field. The data type used for a specific language must be subsequently derived.

This chapter provides the OpenVMS format for all Xlib data structures. The table following each data structure provides a short description of each data structure member. In addition, Table 3–68 provides a list of the error codes that Xlib routines can return.

For a complete description of Xlib data structures and error codes, refer to the *X Window System* book.

### 3.1 Any Event Data Structure

Figure 3–1 illustrates the any event data structure (X\$ANY\_EVENT).

**Figure 3–1 Any Event Data Structure**

|                      |    |
|----------------------|----|
| x\$I_anyv_type       | 0  |
| x\$I_anyv_serial     | 4  |
| x\$I_anyv_send_event | 8  |
| x\$a_anyv_display    | 12 |
| x\$I_anyv_window     | 16 |

## Xlib Data Structures and Error Codes

### 3.1 Any Event Data Structure

Table 3–1 describes the members of the any event data structure.

**Table 3–1 Any Event Data Structure Members**

| Member Name         | Contents                                                                       |
|---------------------|--------------------------------------------------------------------------------|
| XSL_ANYV_TYPE       | Type of event Xlib is reporting                                                |
| XSL_ANYV_SERIAL     | Number of the last request processed by the server                             |
| XSL_ANYV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request |
| XSA_ANYV_DISPLAY    | Display on which the event occurred                                            |
| XSL_ANYV_WINDOW     | Window in which the event occurred                                             |

### 3.2 Arc Data Structure

Figure 3–2 illustrates the arc data structure (X\$ARC).

**Figure 3–2 Arc Data Structure**

|                  |                  |   |
|------------------|------------------|---|
| x\$w_garc_y      | x\$w_garc_x      | 0 |
| x\$w_garc_height | x\$w_garc_width  | 4 |
| x\$w_garc_angle2 | x\$w_garc_angle1 | 8 |

Table 3–2 describes the members of the arc data structure.

**Table 3–2 Arc Data Structure Members**

| Member Name      | Contents                                                                                                      |
|------------------|---------------------------------------------------------------------------------------------------------------|
| X\$W_GARC_X      | Defines the x-coordinate value of the rectangle in which the server draws the arc                             |
| X\$W_GARC_Y      | Defines the y-coordinate value of the rectangle in which the server draws the arc                             |
| X\$W_GARC_WIDTH  | Defines the x axis diameter of the arc                                                                        |
| X\$W_GARC_HEIGHT | Defines the y axis diameter of the arc                                                                        |
| X\$W_GARC_ANGLE1 | Defines the starting point of the arc relative to the three o'clock position from the center of the rectangle |
| X\$W_GARC_ANGLE2 | Defines the extent of the arc relative to the starting point                                                  |

### 3.3 Button Event Data Structure

Figure 3–3 illustrates the button event data structure (X\$BUTTON\_EVENT).

**Figure 3–3 Button Event Data Structure**

|                       |    |
|-----------------------|----|
| x\$I_btev_type        | 0  |
| x\$I_btev_serial      | 4  |
| x\$I_btev_send_event  | 8  |
| x\$a_btev_display     | 12 |
| x\$I_btev_window      | 16 |
| x\$I_btev_root        | 20 |
| x\$I_btev_subwindow   | 24 |
| x\$I_btev_time        | 28 |
| x\$I_btev_x           | 32 |
| x\$I_btev_y           | 36 |
| x\$I_btev_x_root      | 40 |
| x\$I_btev_y_root      | 44 |
| x\$I_btev_state       | 48 |
| x\$I_btev_button      | 52 |
| x\$I_btev_same_screen | 56 |

Table 3–3 describes the members of the button event data structure.

**Table 3–3 Button Event Data Structure Members**

| Member Name         | Contents                                                                                       |
|---------------------|------------------------------------------------------------------------------------------------|
| XSL_BTEV_TYPE       | Type of event reported. The event type can be either x\$C_button_press or x\$C_button_release. |
| XSL_BTEV_SERIAL     | Number of the last request processed by the server.                                            |
| XSL_BTEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                |
| XSA_BTEV_DISPLAY    | Display on which the event occurred.                                                           |
| XSL_BTEV_WINDOW     | Event window.                                                                                  |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.3 Button Event Data Structure

**Table 3–3 (Cont.) Button Event Data Structure Members**

| Member Name          | Contents                                                                                                                                                                                                                                                                                                                                                                                                                                                     |              |              |              |              |              |           |              |           |              |           |
|----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|--------------|--------------|--------------|--------------|-----------|--------------|-----------|--------------|-----------|
| XSL_BTEV_ROOT        | Root window in which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                     |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_SUBWINDOW   | Source window in which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                   |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_TIME        | Time in milliseconds at which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                            |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_X           | The x value of the pointer coordinates in the source window at the time the event occurred.                                                                                                                                                                                                                                                                                                                                                                  |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_Y           | The y value of the pointer coordinates in the source window at the time the event occurred.                                                                                                                                                                                                                                                                                                                                                                  |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_X_ROOT      | The x value of the pointer coordinates, relative to the root window.                                                                                                                                                                                                                                                                                                                                                                                         |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_Y_ROOT      | The y value of the pointer coordinates, relative to the root window.                                                                                                                                                                                                                                                                                                                                                                                         |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_STATE       | State of the button just prior to the event. Xlib can set this member to the bitwise OR of one or more of the following masks:<br><table border="0" style="margin-left: 20px;"> <tr> <td>x\$m_button1</td> <td>x\$m_mod1</td> </tr> <tr> <td>x\$m_button2</td> <td>x\$m_mod2</td> </tr> <tr> <td>x\$m_button3</td> <td>x\$m_mod3</td> </tr> <tr> <td>x\$m_button4</td> <td>x\$m_mod4</td> </tr> <tr> <td>x\$m_button5</td> <td>x\$m_mod5</td> </tr> </table> | x\$m_button1 | x\$m_mod1    | x\$m_button2 | x\$m_mod2    | x\$m_button3 | x\$m_mod3 | x\$m_button4 | x\$m_mod4 | x\$m_button5 | x\$m_mod5 |
| x\$m_button1         | x\$m_mod1                                                                                                                                                                                                                                                                                                                                                                                                                                                    |              |              |              |              |              |           |              |           |              |           |
| x\$m_button2         | x\$m_mod2                                                                                                                                                                                                                                                                                                                                                                                                                                                    |              |              |              |              |              |           |              |           |              |           |
| x\$m_button3         | x\$m_mod3                                                                                                                                                                                                                                                                                                                                                                                                                                                    |              |              |              |              |              |           |              |           |              |           |
| x\$m_button4         | x\$m_mod4                                                                                                                                                                                                                                                                                                                                                                                                                                                    |              |              |              |              |              |           |              |           |              |           |
| x\$m_button5         | x\$m_mod5                                                                                                                                                                                                                                                                                                                                                                                                                                                    |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_BUTTON      | Buttons that changed state. Xlib can set this member to one of the following values:<br><table border="0" style="margin-left: 20px;"> <tr> <td>x\$c_button1</td> <td>x\$c_button4</td> </tr> <tr> <td>x\$c_button2</td> <td>x\$c_button5</td> </tr> <tr> <td>x\$c_button3</td> <td></td> </tr> </table>                                                                                                                                                      | x\$c_button1 | x\$c_button4 | x\$c_button2 | x\$c_button5 | x\$c_button3 |           |              |           |              |           |
| x\$c_button1         | x\$c_button4                                                                                                                                                                                                                                                                                                                                                                                                                                                 |              |              |              |              |              |           |              |           |              |           |
| x\$c_button2         | x\$c_button5                                                                                                                                                                                                                                                                                                                                                                                                                                                 |              |              |              |              |              |           |              |           |              |           |
| x\$c_button3         |                                                                                                                                                                                                                                                                                                                                                                                                                                                              |              |              |              |              |              |           |              |           |              |           |
| XSL_BTEV_SAME_SCREEN | Indicates whether or not the event window is on the same screen as the root window.                                                                                                                                                                                                                                                                                                                                                                          |              |              |              |              |              |           |              |           |              |           |

### 3.4 Char 2B Data Structure

Figure 3–4 illustrates the char 2B data structure (X\$CHAR\_2B).

**Figure 3–4 Char 2B Data Structure**



Table 3–4 describes the members of the char 2B data structure.

**Table 3–4 Char 2B Data Structure Members**

| Member Name       | Contents                              |
|-------------------|---------------------------------------|
| X\$T_CHAR2B_BYTE1 | Row in which the character is indexed |
| X\$T_CHAR2B_BYTE2 | Position of the character in the row  |



### 3.5 Char Struct Data Structure

Figure 3–5 illustrates the char struct data structure (X\$CHAR\_STRUCT).

**Figure 3–5 Char Struct Data Structure**

|                      |                    |   |
|----------------------|--------------------|---|
| x\$w_char_rbearing   | x\$w_char_lbearing | 0 |
| x\$w_char_ascent     | x\$w_char_width    | 4 |
| x\$w_char_attributes | x\$w_char_descent  | 8 |

Table 3–5 describes the members of the char struct data structure.

**Table 3–5 Char Struct Data Structure Members**

| Member Name          | Contents                                                             |
|----------------------|----------------------------------------------------------------------|
| X\$W_CHAR_LBEARING   | Distance from the origin to the left edge of the bounding box.       |
| X\$W_CHAR_RBEARING   | Distance from the origin to the right edge of the bounding box       |
| X\$W_CHAR_WIDTH      | Distance from the current origin to the origin of the next character |
| X\$W_CHAR_ASCENT     | Distance from the baseline to the top of the bounding box            |
| X\$W_CHAR_DESCENT    | Distance from the baseline to the bottom of the bounding box         |
| X\$W_CHAR_ATTRIBUTES | Attributes defined in the bitmap distribution format file            |

### 3.6 Circulate Event Data Structure

Figure 3–6 illustrates the circulate event data structure (X\$CIRCULATE\_EVENT).

**Figure 3–6 Circulate Event Data Structure**

|                      |    |
|----------------------|----|
| x\$I_ciev_type       | 0  |
| x\$I_ciev_serial     | 4  |
| x\$I_ciev_send_event | 8  |
| x\$a_ciev_display    | 12 |
| x\$I_ciev_event      | 16 |
| x\$I_ciev_window     | 20 |
| x\$I_ciev_place      | 24 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.6 Circulate Event Data Structure

Figure 3–6 (Cont.) Circulate Event Data Structure

Table 3–6 describes the members of the circulate event data structure.

**Table 3–6 Circulate Event Data Structure Members**

| Member Name         | Contents                                                                                                 |
|---------------------|----------------------------------------------------------------------------------------------------------|
| XSL_CIEV_TYPE       | Value defined by the constant <code>xSc_circulate_notify</code>                                          |
| XSL_CIEV_SERIAL     | Number of the last request processed by the server                                                       |
| XSL_CIEV_SEND_EVENT | Value defined by the constant <code>true</code> if the event came from a <code>SEND EVENT</code> request |
| XSA_CIEV_DISPLAY    | Display on which the event occurred                                                                      |
| XSL_CIEV_EVENT      | Event window                                                                                             |
| XSL_CIEV_WINDOW     | Window that has been circulated                                                                          |
| XSL_CIEV_PLACE      | Place of the window on the stack after it has been circulated                                            |

### 3.7 Circulate Request Event Data Structure

Figure 3–7 illustrates the circulate request event data structure (`XSCIRC_REQUEST_EVENT`).

Figure 3–7 Circulate Request Event Data Structure

|                                   |    |
|-----------------------------------|----|
| <code>x\$l_cire_type</code>       | 0  |
| <code>x\$l_cire_serial</code>     | 4  |
| <code>x\$l_cire_send_event</code> | 8  |
| <code>x\$a_cire_display</code>    | 12 |
| <code>x\$l_cire_parent</code>     | 16 |
| <code>x\$l_cire_window</code>     | 20 |
| <code>x\$l_cire_place</code>      | 24 |

## Xlib Data Structures and Error Codes

### 3.7 Circulate Request Event Data Structure

Table 3–7 describes the members of the circulate request event data structure.

**Table 3–7 Circulate Request Event Data Structure Members**

| Member Name         | Contents                                                                                                 |
|---------------------|----------------------------------------------------------------------------------------------------------|
| XSL_CIRE_TYPE       | Value defined by the constant <code>xSc_circulate_notify</code>                                          |
| XSL_CIRE_SERIAL     | Number of the last request processed by the server                                                       |
| XSL_CIRE_SEND_EVENT | Value defined by the constant <code>true</code> if the event came from a <code>SEND EVENT</code> request |
| XSA_CIRE_DISPLAY    | Display on which the event occurred                                                                      |
| XSL_CIRE_PARENT     | Parent window                                                                                            |
| XSL_CIRE_WINDOW     | Window that has been circulated                                                                          |
| XSL_CIRE_PLACE      | Place of the window on the stack after it has been circulated                                            |

### 3.8 Class Hint Data Structure

Figure 3–8 illustrates the class hint data structure (`X$CLASS_HINT`).

**Figure 3–8 Class Hint Data Structure**

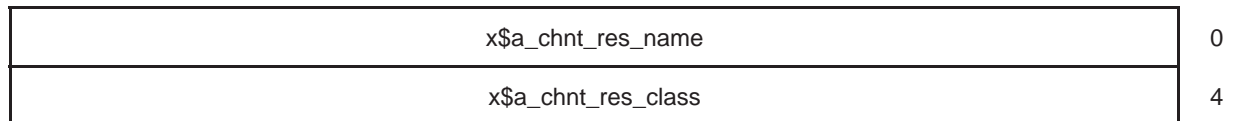


Table 3–8 describes the members of the class hint data structure.

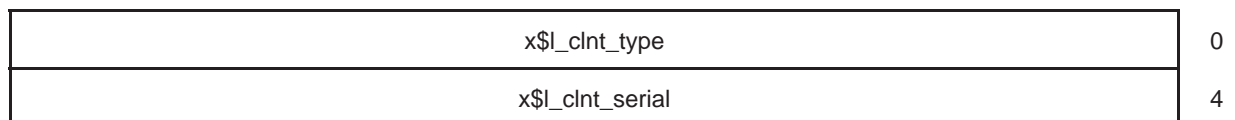
**Table 3–8 Class Hint Data Structure Members**

| Member Name        | Contents                        |
|--------------------|---------------------------------|
| XSA_CHNT_RES_NAME  | Defines the name of the window  |
| XSA_CHNT_RES_CLASS | Defines the class of the window |

### 3.9 Client Message Event Data Structure

Figure 3–9 illustrates the client message event data structure (`X$CLIE_MESSAGE_EVENT`).

**Figure 3–9 Client Message Event Data Structure**



(continued on next page)

## Xlib Data Structures and Error Codes

### 3.9 Client Message Event Data Structure

|                        |    |
|------------------------|----|
| x\$l_clnt_send_event   | 8  |
| x\$a_clnt_display      | 12 |
| x\$l_clnt_window       | 16 |
| x\$l_clnt_message_type | 20 |
| x\$l_clnt_format       | 24 |
| x\$b_clnt_b (20 bytes) | 28 |

Table 3–9 describes the members of the client message event data structure.

**Table 3–9 Client Message Event Data Structure Members**

| Member Name           | Contents                                                                       |
|-----------------------|--------------------------------------------------------------------------------|
| XSL_CLNT_TYPE         | Value defined by the x\$c_client_message constant                              |
| XSL_CLNT_SERIAL       | Number of the last request processed by the server                             |
| XSL_CLNT_SEND_EVENT   | Value defined by the constant true if the event came from a SEND EVENT request |
| X\$a_CLNT_DISPLAY     | Display on which the event occurred                                            |
| XSL_CLNT_WINDOW       | Window to which the message is sent                                            |
| XSL_CLNT_MESSAGE_TYPE | Indicates how the message data is to be interpreted by the receiving client    |
| XSL_CLNT_FORMAT       | Indicates whether the data is in units of 8, 16, or 32 bits                    |
| X\$b_CLNT_B           | Data of twenty 8-bit values.                                                   |

### 3.10 Color Data Structure

Figure 3–10 illustrates the color data structure (X\$COLOR).

**Figure 3–10 Color Data Structure**

|                 |                 |                |  |   |
|-----------------|-----------------|----------------|--|---|
| x\$l_colr_pixel |                 |                |  | 0 |
| x\$w_colr_green |                 | x\$w_colr_red  |  | 4 |
| x\$b_colr_pad   | x\$b_colr_flags | x\$w_colr_blue |  | 8 |

Table 3–10 describes the members of the color data structure.

**Table 3–10 Color Data Structure Members**

| Member Name    | Contents                                                                                                                                       |
|----------------|------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_COLR_PIXEL | Defines a pixel value.                                                                                                                         |
| XSW_COLR_RED   | Defines the red value of the pixel. <sup>1</sup>                                                                                               |
| XSW_COLR_GREEN | Defines the green value of the pixel. <sup>1</sup>                                                                                             |
| XSW_COLR_BLUE  | Defines the blue value of the pixel. <sup>1</sup>                                                                                              |
| XSB_COLR_FLAGS | Defines which color members are to be defined in the color map. Possible flags are as follows:<br>x\$m_do_red<br>x\$m_do_green<br>x\$m_do_blue |
| XSB_COLR_PAD   | Makes the structure an even length.                                                                                                            |

<sup>1</sup>Color values range from 0 to 65,535. “On full” in a color is a value of 65,535, independent of the number of planes of the display. Half brightness in a color is a value of 32,767; off is a value of 0. This representation gives uniform results for color values across displays with different color resolution.

### 3.11 Color Map Event Data Structure

Figure 3–11 illustrates the color map event data structure (X\$COLORMAP\_EVENT).

**Figure 3–11 Color Map Event Data Structure**

|                      |    |
|----------------------|----|
| x\$I_cmev_type       | 0  |
| x\$I_cmev_serial     | 4  |
| x\$I_cmev_send_event | 8  |
| x\$a_cmev_display    | 12 |
| x\$I_cmev_window     | 16 |
| x\$I_cmev_colormap   | 20 |
| x\$I_cmev_new        | 24 |
| x\$I_cmev_state      | 28 |

## Xlib Data Structures and Error Codes

### 3.11 Color Map Event Data Structure

Table 3–11 describes the members of the color map event data structure.

**Table 3–11 Color Map Event Data Structure Members**

| Member Name         | Contents                                                                                                                                                                                                                                                                                                                                                |
|---------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_CMEV_TYPE       | Value defined by the xSc_colormap_notify constant.                                                                                                                                                                                                                                                                                                      |
| XSL_CMEV_SERIAL     | Number of the last request processed by the server.                                                                                                                                                                                                                                                                                                     |
| XSL_CMEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                                                                                                                                                                                                                                                                         |
| XSA_CMEV_DISPLAY    | Display on which the event occurred.                                                                                                                                                                                                                                                                                                                    |
| XSL_CMEV_WINDOW     | Window whose associated color map has changed.                                                                                                                                                                                                                                                                                                          |
| XSL_CMEV_COLORMAP   | If the window manager changes the color map in response to a call to CHANGE WINDOW ATTRIBUTES, INSTALL COLORMAP, or UNINSTALL COLORMAP, this member has a value specified by the constant xSc_colormap. If the window manager changes the color map in response to a call to FREE COLORMAP, this member has a value specified by the constant xSc_none. |
| XSL_CMEV_NEW        | Value defined by the constant true if the window manager has changed the color map or the value defined by the constant false if the window manager has installed or removed the color map.                                                                                                                                                             |
| XSL_CMEV_STATE      | Value defined by the constant xSc_colormap_installed if the color map is installed. The value defined by the constant xSc_colormap_uninstalled if the color map is not installed.                                                                                                                                                                       |

### 3.12 Compose Status Data Structure

Figure 3–12 illustrates the compose status data structure (X\$COMPOSE\_STATUS).

**Figure 3–12 Compose Status Data Structure**

|                         |   |
|-------------------------|---|
| x\$a_cmps_compose_ptr   | 0 |
| x\$l_cmps_chars_matched | 4 |

Table 3–12 describes the members of the compose status data structure.

**Table 3–12 Compose Data Structure Members**

| Member Name            | Contents                    |
|------------------------|-----------------------------|
| XSA_CMPS_COMPOSE_PTR   | Compose state table pointer |
| XSL_CMPS_CHARS_MATCHED | Characters match state      |

### 3.13 Configure Event Data Structure

Figure 3–13 illustrates the configure event data structure (X\$CONFIGURE\_EVENT).

Figure 3–13 Configure Event Data Structure

|                             |    |
|-----------------------------|----|
| x\$I_cfev_type              | 0  |
| x\$I_cfev_serial            | 4  |
| x\$I_cfev_send_event        | 8  |
| x\$a_cfev_display           | 12 |
| x\$I_cfev_event             | 16 |
| x\$I_cfev_window            | 20 |
| x\$I_cfev_x                 | 24 |
| x\$I_cfev_y                 | 28 |
| x\$I_cfev_width             | 32 |
| x\$I_cfev_height            | 36 |
| x\$I_cfev_border_width      | 40 |
| x\$I_cfev_above             | 44 |
| x\$I_cfev_override_redirect | 48 |

Table 3–13 describes the members of the configure event data structure.

Table 3–13 Configure Event Data Structure Members

| Member Name         | Contents                                                                                                                               |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| XSL_CFEV_TYPE       | Value defined by the constant xSc_cfev_configure_notify.                                                                               |
| XSL_CFEV_SERIAL     | Number of the last request processed by the server.                                                                                    |
| XSL_CFEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                                                        |
| XSA_CFEV_DISPLAY    | Display on which the event occurred.                                                                                                   |
| XSL_CFEV_EVENT      | Event window.                                                                                                                          |
| XSL_CFEV_WINDOW     | Window that has been reconfigured.                                                                                                     |
| XSL_CFEV_X          | The x value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window. |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.13 Configure Event Data Structure

**Table 3–13 (Cont.) Configure Event Data Structure Members**

| Member Name                | Contents                                                                                                                                                                                            |
|----------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_CFEV_Y                 | The y value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window.                                                              |
| XSL_CFEV_WIDTH             | Width of the window, excluding the border.                                                                                                                                                          |
| XSL_CFEV_HEIGHT            | Height of the window, excluding the border.                                                                                                                                                         |
| XSL_CFEV_BORDER_WIDTH      | Border width of the reconfigured window.                                                                                                                                                            |
| XSL_CFEV_ABOVE             | Identifier of the sibling window above which the window is stacked. If this member has a value specified by the constant <code>xSc_none</code> , Xlib places the window at the bottom of the stack. |
| XSL_CFEV_OVERRIDE_REDIRECT | If this member has a value defined by the constant <code>true</code> , the window manager ignores requests to reconfigure the window.                                                               |

### 3.14 Configure Request Event Data Structure

Figure 3–14 illustrates the configure request event data structure (`X$CONF_REQUEST_EVENT`).

**Figure 3–14 Configure Event Data Structure**

|                        |    |
|------------------------|----|
| x\$l_cfre_type         | 0  |
| x\$l_cfre_serial       | 4  |
| x\$l_cfre_send_event   | 8  |
| x\$a_cfre_display      | 12 |
| x\$l_cfre_parent       | 16 |
| x\$l_cfre_window       | 20 |
| x\$l_cfre_x            | 24 |
| x\$l_cfre_y            | 28 |
| x\$l_cfre_width        | 32 |
| x\$l_cfre_height       | 36 |
| x\$l_cfre_border_width | 40 |
| x\$l_cfre_above        | 44 |
| x\$l_cfre_detail       | 48 |
| x\$l_cfre_value_mask   | 52 |

(continued on next page)



## Xlib Data Structures and Error Codes

### 3.14 Configure Request Event Data Structure

**Figure 3–14 (Cont.) Configure Event Data Structure**

Table 3–14 describes the members of the configure request event data structure.

**Table 3–14 Configure Request Event Data Structure Members**

| Member Name               | Contents                                                                                                                                                                                                                                                                                                                                                                                                                   |                        |                        |                         |                            |                           |  |
|---------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------|------------------------|-------------------------|----------------------------|---------------------------|--|
| XSL_CFRE_TYPE             | Value defined by the constant <code>xSc_cfev_configure_notify</code> .                                                                                                                                                                                                                                                                                                                                                     |                        |                        |                         |                            |                           |  |
| XSL_CFRE_SERIAL           | Number of the last request processed by the server.                                                                                                                                                                                                                                                                                                                                                                        |                        |                        |                         |                            |                           |  |
| XSL_CFRE_SEND_EVENT       | Value defined by the constant <code>true</code> if the event came from a <code>SEND_EVENT</code> request.                                                                                                                                                                                                                                                                                                                  |                        |                        |                         |                            |                           |  |
| XSA_CFRE_DISPLAY          | Display on which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                       |                        |                        |                         |                            |                           |  |
| XSL_CFRE_PARENT           | Parent window.                                                                                                                                                                                                                                                                                                                                                                                                             |                        |                        |                         |                            |                           |  |
| XSL_CFRE_WINDOW           | Window that has been reconfigured.                                                                                                                                                                                                                                                                                                                                                                                         |                        |                        |                         |                            |                           |  |
| XSL_CFRE_X                | The x value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window.                                                                                                                                                                                                                                                                                     |                        |                        |                         |                            |                           |  |
| XSL_CFRE_Y                | The y value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window.                                                                                                                                                                                                                                                                                     |                        |                        |                         |                            |                           |  |
| XSL_CFRE_WIDTH            | Width of the window, excluding the border.                                                                                                                                                                                                                                                                                                                                                                                 |                        |                        |                         |                            |                           |  |
| XSL_CFRE_HEIGHT           | Height of the window, excluding the border.                                                                                                                                                                                                                                                                                                                                                                                |                        |                        |                         |                            |                           |  |
| XSL_CFRE_BORDER_WIDTH     | Border width of the reconfigured window.                                                                                                                                                                                                                                                                                                                                                                                   |                        |                        |                         |                            |                           |  |
| XSL_CFRE_ABOVE            | Identifier of the sibling window above which the window is stacked. If this member has a value specified by the constant <code>xSc_none</code> , Xlib places the window at the bottom of the stack.                                                                                                                                                                                                                        |                        |                        |                         |                            |                           |  |
| XSL_CFRE_DETAIL           | Indicates which windows were specified in the protocol request.                                                                                                                                                                                                                                                                                                                                                            |                        |                        |                         |                            |                           |  |
| XSL_CFRE_VALUE_MASK       | Components that were specified in the <code>configure_protocol</code> request. Xlib can specify in this member one of the following constants: <table style="margin-left: 2em; border: none;"> <tr> <td><code>xSc_above</code></td> <td><code>xSc_below</code></td> </tr> <tr> <td><code>xSc_top_if</code></td> <td><code>xSc_bottom_if</code></td> </tr> <tr> <td><code>xSc_opposite</code></td> <td></td> </tr> </table> | <code>xSc_above</code> | <code>xSc_below</code> | <code>xSc_top_if</code> | <code>xSc_bottom_if</code> | <code>xSc_opposite</code> |  |
| <code>xSc_above</code>    | <code>xSc_below</code>                                                                                                                                                                                                                                                                                                                                                                                                     |                        |                        |                         |                            |                           |  |
| <code>xSc_top_if</code>   | <code>xSc_bottom_if</code>                                                                                                                                                                                                                                                                                                                                                                                                 |                        |                        |                         |                            |                           |  |
| <code>xSc_opposite</code> |                                                                                                                                                                                                                                                                                                                                                                                                                            |                        |                        |                         |                            |                           |  |

### 3.15 Create Window Event Data Structure

Figure 3–15 illustrates the create window event data structure (`XSCREAT_WINDOW_EVENT`).

## Xlib Data Structures and Error Codes

### 3.15 Create Window Event Data Structure

Figure 3–15 Create Window Event Data Structure

|                            |    |
|----------------------------|----|
| x\$_cwev_type              | 0  |
| x\$_cwev_serial            | 4  |
| x\$_cwev_send_event        | 8  |
| x\$a_cwev_display          | 12 |
| x\$_cwev_parent            | 16 |
| x\$_cwev_window            | 20 |
| x\$_cwev_x                 | 24 |
| x\$_cwev_y                 | 28 |
| x\$_cwev_width             | 32 |
| x\$_cwev_height            | 36 |
| x\$_cwev_border_width      | 40 |
| x\$_cwev_override_redirect | 44 |

Table 3–15 describes the members of the create window event data structure.

**Table 3–15 Create Window Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_CWEV_TYPE       | Value defined by the constant x\$C_create_notify.                               |
| XSL_CWEV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_CWEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| XSA_CWEV_DISPLAY    | Display on which the event occurred.                                            |
| XSL_CWEV_PARENT     | Parent window.                                                                  |
| XSL_CWEV_WINDOW     | Window that has been created.                                                   |
| XSL_CWEV_X          | The x value of the coordinates that define the origin of the window.            |
| XSL_CWEV_Y          | The y value of the coordinates that define the origin of the window.            |
| XSL_CWEV_WIDTH      | Width of the newly created window, excluding the border.                        |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.15 Create Window Event Data Structure

**Table 3–15 (Cont.) Create Window Event Data Structure Members**

| Member Name                | Contents                                                                                                           |
|----------------------------|--------------------------------------------------------------------------------------------------------------------|
| XSL_CWEV_HEIGHT            | Height of the newly created window, excluding the border.                                                          |
| XSL_CWEV_BORDER_WIDTH      | Border width of the new window.                                                                                    |
| XSL_CWEV_OVERRIDE_REDIRECT | If this member has a value defined by the constant true, the window manager ignores requests to create the window. |

### 3.16 Crossing Event Data Structure

Figure 3–16 illustrates the crossing event data structure (XCROSSING\_EVENT).

**Figure 3–16 Crossing Event Data Structure**

|                        |    |
|------------------------|----|
| x\$_l_crev_type        | 0  |
| x\$_l_crev_serial      | 4  |
| x\$_l_crev_send_event  | 8  |
| x\$_a_crev_display     | 12 |
| x\$_l_crev_window      | 16 |
| x\$_l_crev_root        | 20 |
| x\$_l_crev_subwindow   | 24 |
| x\$_l_crev_time        | 28 |
| x\$_l_crev_x           | 32 |
| x\$_l_crev_y           | 36 |
| x\$_l_crev_x_root      | 40 |
| x\$_l_crev_y_root      | 44 |
| x\$_l_crev_mode        | 48 |
| x\$_l_crev_detail      | 52 |
| x\$_l_crev_same_screen | 56 |
| x\$_l_crev_focus       | 60 |
| x\$_l_crev_state       | 64 |

## Xlib Data Structures and Error Codes

### 3.16 Crossing Event Data Structure

Table 3–16 describes the members of the crossing event data structure.

**Table 3–16 Crossing Event Data Structure Members**

| Member Name                               | Contents                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
|-------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------|---------------------------------|----------------------------------|-----------------------------------|-------------------------------------------|------------------------|---------------------------|-------------------------|---------------------------|------------------------|------------------------|---------------------------|------------------------|--|
| XSL_CREV_TYPE                             | Value defined by either the <code>xSc_enter_notify</code> or the <code>xSc_leave_notify</code> constant.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_SERIAL                           | Number of the last request processed by the server.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_SEND_EVENT                       | Value defined by the constant <code>true</code> if the event came from a <code>SEND_EVENT</code> request.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSA_CREV_DISPLAY                          | Display on which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_WINDOW                           | Event window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_ROOT                             | Root window in which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_SUBWINDOW                        | Source window in which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_TIME                             | Time in milliseconds at which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_X                                | The x value of the pointer coordinates in the source window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_Y                                | The y value of the pointer coordinates in the source window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_X_ROOT                           | The x value of the pointer coordinates relative to the root window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_Y_ROOT                           | The y value of the pointer coordinates relative to the root window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_MODE                             | Indicates whether the event is normal or pseudomotion. Xlib can set this member to the value specified by <code>xSc_notify_normal</code> , <code>xSc_notify_grab</code> , and <code>xSc_notify_ungrab</code> .                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_DETAIL                           | Indicates which windows Xlib notifies of the window entry or exit event. Xlib can specify in this member one of the following constants:<br><table data-bbox="665 1150 1344 1260"> <tr> <td><code>xSc_notify_ancestor</code></td> <td><code>xSc_notify_virtual</code></td> </tr> <tr> <td><code>xSc_notify_inferior</code></td> <td><code>xSc_notify_nonlinear</code></td> </tr> <tr> <td><code>xSc_notify_nonlinear_virtual</code></td> <td></td> </tr> </table>                                                                                                                                                                                                                        | <code>xSc_notify_ancestor</code> | <code>xSc_notify_virtual</code> | <code>xSc_notify_inferior</code> | <code>xSc_notify_nonlinear</code> | <code>xSc_notify_nonlinear_virtual</code> |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>xSc_notify_ancestor</code>          | <code>xSc_notify_virtual</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>xSc_notify_inferior</code>          | <code>xSc_notify_nonlinear</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>xSc_notify_nonlinear_virtual</code> |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_SAME_SCREEN                      | Indicates whether or not the event window is on the same screen as the root window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_FOCUS                            | Specifies whether the event window or an inferior is the focus window. If true, the event window is the focus window. If false, an inferior is the focus window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| XSL_CREV_STATE                            | State of buttons and keys just prior to the event. Xlib can return the following constants:<br><table data-bbox="665 1495 1117 1759"> <tr> <td><code>x\$m_button1</code></td> <td><code>x\$m_mod3</code></td> </tr> <tr> <td><code>x\$m_button2</code></td> <td><code>x\$m_mod4</code></td> </tr> <tr> <td><code>x\$m_button3</code></td> <td><code>x\$m_mod5</code></td> </tr> <tr> <td><code>x\$m_button4</code></td> <td><code>x\$m_shift</code></td> </tr> <tr> <td><code>x\$m_button5</code></td> <td><code>x\$m_lock</code></td> </tr> <tr> <td><code>x\$m_mod1</code></td> <td><code>x\$m_control</code></td> </tr> <tr> <td><code>x\$m_mod2</code></td> <td></td> </tr> </table> | <code>x\$m_button1</code>        | <code>x\$m_mod3</code>          | <code>x\$m_button2</code>        | <code>x\$m_mod4</code>            | <code>x\$m_button3</code>                 | <code>x\$m_mod5</code> | <code>x\$m_button4</code> | <code>x\$m_shift</code> | <code>x\$m_button5</code> | <code>x\$m_lock</code> | <code>x\$m_mod1</code> | <code>x\$m_control</code> | <code>x\$m_mod2</code> |  |
| <code>x\$m_button1</code>                 | <code>x\$m_mod3</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>x\$m_button2</code>                 | <code>x\$m_mod4</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>x\$m_button3</code>                 | <code>x\$m_mod5</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>x\$m_button4</code>                 | <code>x\$m_shift</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>x\$m_button5</code>                 | <code>x\$m_lock</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>x\$m_mod1</code>                    | <code>x\$m_control</code>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |
| <code>x\$m_mod2</code>                    |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                                  |                                 |                                  |                                   |                                           |                        |                           |                         |                           |                        |                        |                           |                        |  |

### 3.17 Destroy Window Event Data Structure

Figure 3–17 illustrates the destroy window event data structure (X\$DESTR\_WINDOW\_EVENT).

**Figure 3–17 Destroy Window Event Data Structure**

|                     |    |
|---------------------|----|
| x\$_dwev_type       | 0  |
| x\$_dwev_serial     | 4  |
| x\$_dwev_send_event | 8  |
| x\$a_dwev_display   | 12 |
| x\$_dwev_event      | 16 |
| x\$_dwev_window     | 20 |

Table 3–17 describes the members of the destroy window event data structure.

**Table 3–17 Destroy Window Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_DWEV_TYPE       | Value defined by the x\$c_destroy_notify constant.                              |
| XSL_DWEV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_DWEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| X\$a_DWEV_DISPLAY   | Display on which the event occurred.                                            |
| XSL_DWEV_EVENT      | Event window.                                                                   |
| XSL_DWEV_WINDOW     | Window that has been destroyed.                                                 |

### 3.18 Error Event Data Structure

Figure 3–18 illustrates the error event data structure (X\$ERROR\_EVENT).

## Xlib Data Structures and Error Codes

### 3.18 Error Event Data Structure

Figure 3–18 Error Event Data Structure

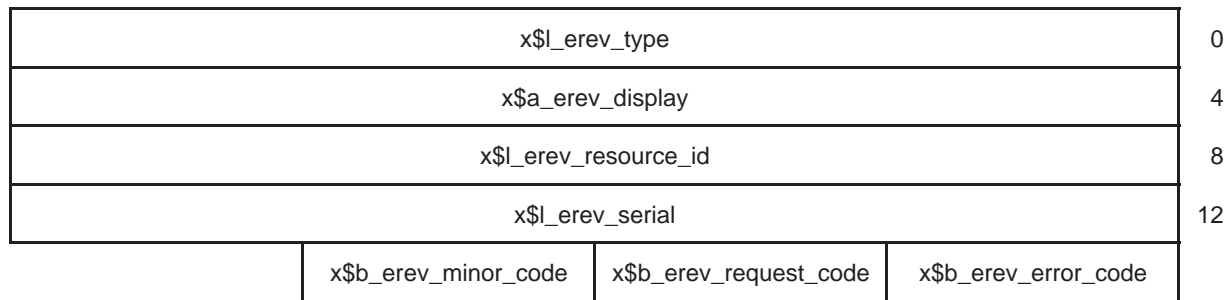


Table 3–18 describes the members of the error event data structure.

Table 3–18 Error Event Data Structure Members

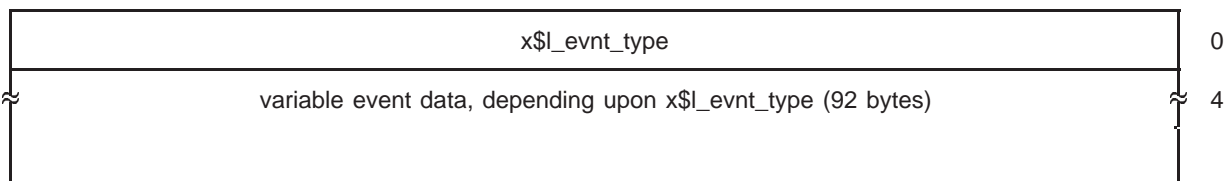
| Member Name           | Contents                                                                                                                                                                                                 |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_EREV_TYPE         | The type of event.                                                                                                                                                                                       |
| XSA_EREV_DISPLAY      | The display information originally returned by OPEN DISPLAY.                                                                                                                                             |
| XSL_EREV_RESOURCE_ID  | The resource identifier associated with the error.                                                                                                                                                       |
| XSL_EREV_SERIAL       | The serial number of the request processed by the server immediately before the failing call was made. Xlib and the server use the serial number to identify the protocol request that caused the error. |
| XSB_EREV_ERROR_CODE   | The error code of the failed request. The error codes are described in Table 3–68.                                                                                                                       |
| XSB_EREV_REQUEST_CODE | The major operation code of the failed request.                                                                                                                                                          |
| XSB_EREV_MINOR_CODE   | The minor operation code of the failed request.                                                                                                                                                          |

See Table 3–68 for a list of Xlib error codes.

### 3.19 Event Data Structure

Figure 3–19 illustrates the event data structure (XSEVENT).

Figure 3–19 Event Data Structure



For the format of the members of the event data structure, see the section in this chapter that details the specific event.

### 3.19.1 The Event Mask

The event routines require that you use the **event\_mask** argument to define the events for which a client receives notice. To do this, you pass an event mask to an Xlib event-handling function that takes an event\_mask argument. The event mask name describes the events that you want the server to return to the client.

Table 3–19 lists the elements of the event mask and their descriptions.

**Table 3–19 Event Mask Elements**

| Bit | OpenVMS Value             | Description                                                   |
|-----|---------------------------|---------------------------------------------------------------|
| 1   | XSM_KEY_PRESS             | Keyboard down events wanted                                   |
| 2   | XSM_KEY_RELEASE           | Keyboard up events wanted                                     |
| 3   | XSM_BUTTON_PRESS          | Pointer button down events wanted                             |
| 4   | XSM_BUTTON_RELEASE        | Pointer button up events wanted                               |
| 5   | XSM_ENTER_WINDOW          | Pointerwindow entry events wanted                             |
| 6   | XSM_LEAVE_WINDOW          | Pointerwindow leave events wanted                             |
| 7   | XSM_POINTER_MOTION        | Pointer motion events wanted                                  |
| 8   | XSM_POINTER_MOTION_HINT   | Pointer motion hints wanted                                   |
| 9   | XSM_BUTTON1_MOTION        | Pointer motion while button 1 down                            |
| 10  | XSM_BUTTON2_MOTION        | Pointer motion while button 2 down                            |
| 11  | XSM_BUTTON3_MOTION        | Pointer motion while button 3 down                            |
| 12  | XSM_BUTTON4_MOTION        | Pointer motion while button 4 down                            |
| 13  | XSM_BUTTON5_MOTION        | Pointer motion while button 5 down                            |
| 14  | XSM_BUTTON_MOTION         | Pointer motion while any button down                          |
| 15  | XSM_KEYMAP_STATE          | Keyboard state wanted at window entry and focus in            |
| 16  | XSM_EXPOSURE              | Any exposure wanted                                           |
| 17  | XSM_VISIBILITY_CHANGE     | Any change in visibility wanted                               |
| 18  | XSM_STRUCTURE_NOTIFY      | Any change in window structure wanted                         |
| 19  | XSM_RESIZE_REDIRECT       | Redirect resize of this window                                |
| 20  | XSM_SUBSTRUCTURE_NOTIFY   | Substructure notification wanted                              |
| 21  | XSM_SUBSTRUCTURE_REDIRECT | Redirect substructure of window                               |
| 22  | XSM_FOCUS_CHANGE          | Any change in input focus wanted                              |
| 23  | XSM_PROPERTY_CHANGE       | Any change in property wanted                                 |
| 24  | XSM_COLORMAP_CHANGE       | Any change in color map wanted                                |
| 25  | XSM_OWNER_GRAB_BUTTON     | Automatic grabs should activate with owner_events set to true |

You can also specify XSM\_NO\_EVENT to indicate that no events are wanted.

### 3.20 Expose Event Data Structure

Figure 3–20 illustrates the expose event data structure (X\$EXPOSE\_EVENT).

## Xlib Data Structures and Error Codes

### 3.20 Expose Event Data Structure

Figure 3–20 Expose Event Data Structure

|                      |    |
|----------------------|----|
| x\$l_exev_type       | 0  |
| x\$l_exev_serial     | 4  |
| x\$l_exev_send_event | 8  |
| x\$a_exev_display    | 12 |
| x\$l_exev_window     | 16 |
| x\$l_exev_x          | 20 |
| x\$l_exev_y          | 24 |
| x\$l_exev_width      | 28 |
| x\$l_exev_height     | 32 |
| x\$l_exev_count      | 36 |

Table 3–20 describes the members of the expose event data structure.

**Table 3–20 Expose Event Data Structure Members**

| Member Name         | Contents                                                                               |
|---------------------|----------------------------------------------------------------------------------------|
| XSL_EXEV_TYPE       | Value defined by the x\$C_expose constant                                              |
| XSL_EXEV_SERIAL     | Number of the last request processed by the server                                     |
| XSL_EXEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request         |
| XSA_EXEV_DISPLAY    | Display on which the event occurred                                                    |
| XSL_EXEV_WINDOW     | Event window                                                                           |
| XSL_EXEV_X          | The x value of the coordinates that define the upper left corner of the exposed region |
| XSL_EXEV_Y          | The y value of the coordinates that define the upper left corner of the exposed region |
| XSL_EXEV_WIDTH      | Width of the exposed region                                                            |
| XSL_EXEV_HEIGHT     | Height of the exposed region                                                           |
| XSL_EXEV_COUNT      | Number of exposure events that are to follow                                           |



### 3.21 Ext Codes Data Structure

Figure 3–21 illustrates the ext codes data structure (XSEXT\_CODES).

**Figure 3–21 Ext Codes Data Structure**

|                        |    |
|------------------------|----|
| x\$l_extc_extension    | 0  |
| x\$l_extc_major_opcode | 4  |
| x\$l_extc_first_event  | 8  |
| x\$l_extc_first_error  | 12 |

Table 3–21 describes the members of the ext codes data structure (XSEXT\_CODES).

**Table 3–21 Ext Codes Data Structure**

| Member Name           | Contents                             |
|-----------------------|--------------------------------------|
| XSL_EXTC_EXTENSION    | Extension number                     |
| XSL_EXTC_MAJOR_OPCODE | Major op-code assigned by server     |
| XSL_EXTC_FIRST_EVENT  | First event number for the extension |
| XSL_EXTC_FIRST_ERROR  | First error number for the extension |

### 3.22 Ext Data Data Structure

Figure 3–22 illustrates the ext data data structure (XSEXT\_DATA).

**Figure 3–22 Ext Codes Data Structure**

|                   |    |
|-------------------|----|
| x\$l_extd_number  | 0  |
| x\$a_extd_next    | 4  |
| x\$a_extd_free    | 8  |
| x\$l_extd_private | 12 |

## Xlib Data Structures and Error Codes

### 3.22 Ext Data Data Structure

Table 3–22 describes the members of the ext data data structure.

**Table 3–22 Ext Data Data Structure**

| Member Name      | Contents                                |
|------------------|-----------------------------------------|
| XSL_EXTD_NUMBER  | Number returned by X\$INIT_EXTENSION    |
| XSA_EXTD_NEXT    | Next item on list of data for structure |
| XSA_EXTD_FREE    | Called to free private, if defined      |
| XSL_EXTD_PRIVATE | Data private to this extension          |

### 3.23 Focus Change Event Data Structure

Figure 3–23 illustrates the focus change event data structure (X\$FOCUS\_CHANGE\_EVENT).

**Figure 3–23 Focus Change Event Data Structure**

|                      |    |
|----------------------|----|
| x\$I_fcev_type       | 0  |
| x\$I_fcev_serial     | 4  |
| x\$I_fcev_send_event | 8  |
| x\$a_fcev_display    | 12 |
| x\$I_fcev_window     | 16 |
| x\$I_fcev_mode       | 20 |
| x\$I_fcev_detail     | 24 |

Table 3–23 describes the members of the focus change event data structure.

**Table 3–23 Focus Change Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_FCEV_TYPE       | Value defined by either the x\$C_focus_in or x\$C_focus_out constant.           |
| XSL_FCEV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_FCEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| XSA_FCEV_DISPLAY    | Display on which the event occurred.                                            |
| XSL_FCEV_WINDOW     | Event window.                                                                   |

(continued on next page)

**Table 3–23 (Cont.) Focus Change Event Data Structure Members**

| Member Name                   | Contents                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                      |                           |                      |                       |                               |                     |                          |                         |
|-------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------|---------------------------|----------------------|-----------------------|-------------------------------|---------------------|--------------------------|-------------------------|
| XSL_FCEV_MODE                 | <p>Specifies whether the event is the result of normal keyboard input, keyboard input after a client has grabbed the keyboard, keyboard input at the time the client activates a keyboard grab, or keyboard input at the time the client deactivates a keyboard grab. Xlib can set this field to one of the following constants:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">x\$C_notify_normal</td> <td style="width: 50%;">x\$C_notify_while_grabbed</td> </tr> <tr> <td>x\$C_notify_grab</td> <td>x\$C_notify_ungrab</td> </tr> </table> | x\$C_notify_normal   | x\$C_notify_while_grabbed | x\$C_notify_grab     | x\$C_notify_ungrab    |                               |                     |                          |                         |
| x\$C_notify_normal            | x\$C_notify_while_grabbed                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                      |                           |                      |                       |                               |                     |                          |                         |
| x\$C_notify_grab              | x\$C_notify_ungrab                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                      |                           |                      |                       |                               |                     |                          |                         |
| XSL_FCEV_DETAIL               | <p>Specifies which windows and pointers Xlib notifies of the input focus change. Xlib can set this field to one of the following constants:</p> <table style="width: 100%; border: none;"> <tr> <td style="width: 50%;">x\$C_notify_ancestor</td> <td style="width: 50%;">x\$C_notify_virtual</td> </tr> <tr> <td>x\$C_notify_inferior</td> <td>x\$C_notify_nonlinear</td> </tr> <tr> <td>x\$C_notify_nonlinear_virtual</td> <td>x\$C_notify_pointer</td> </tr> <tr> <td>x\$C_notify_pointer_root</td> <td>x\$C_notify_detail_none</td> </tr> </table>                      | x\$C_notify_ancestor | x\$C_notify_virtual       | x\$C_notify_inferior | x\$C_notify_nonlinear | x\$C_notify_nonlinear_virtual | x\$C_notify_pointer | x\$C_notify_pointer_root | x\$C_notify_detail_none |
| x\$C_notify_ancestor          | x\$C_notify_virtual                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                      |                           |                      |                       |                               |                     |                          |                         |
| x\$C_notify_inferior          | x\$C_notify_nonlinear                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |                      |                           |                      |                       |                               |                     |                          |                         |
| x\$C_notify_nonlinear_virtual | x\$C_notify_pointer                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                      |                           |                      |                       |                               |                     |                          |                         |
| x\$C_notify_pointer_root      | x\$C_notify_detail_none                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                      |                           |                      |                       |                               |                     |                          |                         |

### 3.24 Font Property Data Structure

Figure 3–24 illustrates the font property data structure (X\$FONT\_PROP).

**Figure 3–24 Font Property Data Structure**

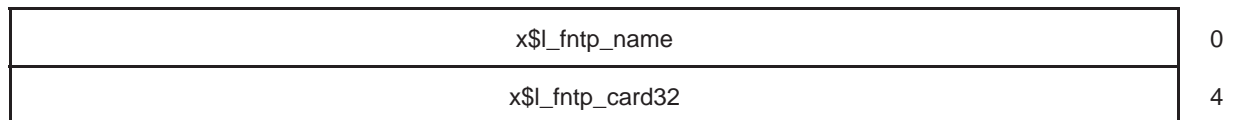


Table 3–24 describes the members of the font property data structure.

**Table 3–24 Font Property Data Structure Members**

| Member Name     | Contents                                         |
|-----------------|--------------------------------------------------|
| XSL_FNTP_NAME   | The string of characters that names the property |
| XSL_FNTP_CARD32 | A 32-bit value that defines the font property    |

## Xlib Data Structures and Error Codes

### 3.25 Font Data Structure

### 3.25 Font Data Structure

Figure 3–25 illustrates the font data structure (XSFONT\_STRUCT).

Figure 3–25 Font Data Structure

|                             |    |
|-----------------------------|----|
| x\$a_fstr_ext_data          | 0  |
| x\$l_fstr_fid               | 4  |
| x\$l_fstr_direction         | 8  |
| x\$l_fstr_min_char_or_byte2 | 12 |
| x\$l_fstr_max_char_or_byte2 | 16 |
| x\$l_fstr_min_byte1         | 20 |
| x\$l_fstr_max_byte1         | 24 |
| x\$l_fstr_all_chars_exist   | 28 |
| x\$l_fstr_default_char      | 32 |
| x\$l_fstr_n_properties      | 36 |
| x\$a_fstr_properties        | 40 |
| x\$a_fstr_min_bounds        | 44 |
| x\$a_fstr_max_bounds        | 48 |
| x\$a_fstr_per_char          | 52 |
| x\$l_fstr_ascent            | 56 |
| x\$l_fstr_descent           | 60 |

Table 3–25 describes the members of the font data structure.

**Table 3–25 Font Data Structure Members**

| Member Name                | Contents                                                                                                                                   |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------|
| XSA_FSTR_EXT_DATA          | Data used by extensions.                                                                                                                   |
| XSL_FSTR_FID               | Identifier of the font.                                                                                                                    |
| XSL_FSTR_DIRECTION         | Hint about the direction in which the font is painted.                                                                                     |
| XSL_FSTR_MIN_CHAR_OR_BYTE2 | The first character in the font.                                                                                                           |
| XSL_FSTR_MAX_CHAR_OR_BYTE2 | The last character in the font.                                                                                                            |
| XSL_FSTR_MIN_BYTE1         | First existing row.                                                                                                                        |
| XSL_FSTR_MAX_BYTE1         | Last existing row.                                                                                                                         |
| XSL_FSTR_ALL_CHARS_EXIST   | If the value of this member is true, all characters in the array pointed to by XSA_FSTR_PER_CHAR have nonzero bounding boxes.              |
| XSL_FSTR_DEFAULT_CHAR      | The character used when an undefined or nonexistent character is specified.                                                                |
| XSL_FSTR_N_PROPERTIES      | The number of properties associated with the font.                                                                                         |
| XSA_FSTR_PROPERTIES        | The address of an array of font property structures that define font properties.                                                           |
| XSR_FSTR_MIN_BOUNDS        | The minimum metrics values of all the characters in the font.                                                                              |
| XSR_FSTR_MAX_BOUNDS        | The maximum metrics values of all the characters in the font.                                                                              |
| XSA_FSTR_PER_CHAR          | The address of an array of character structures that define each character in the font. For a fixed font the value of this member is null. |
| XSL_FSTR_ASCENT            | The distance from the baseline to the top of the bounding box.                                                                             |
| XSL_FSTR_DESCENT           | The distance from the baseline to the bottom of the bounding box.                                                                          |

### 3.26 The GC Values Data Structure

Figure 3–26 illustrates the GC values data structure (X\$GC\_VALUES).

# Xlib Data Structures and Error Codes

## 3.26 The GC Values Data Structure

Figure 3–26 GC Values Data Structure

|                               |    |
|-------------------------------|----|
| x\$_l_gcvl_function           | 0  |
| x\$_l_gcvl_plane_mask         | 4  |
| x\$_l_gcvl_foreground         | 8  |
| x\$_l_gcvl_background         | 12 |
| x\$_l_gcvl_line_width         | 16 |
| x\$_l_gcvl_line_style         | 20 |
| x\$_l_gcvl_cap_style          | 24 |
| x\$_l_gcvl_join_style         | 28 |
| x\$_l_gcvl_fill_style         | 32 |
| x\$_l_gcvl_fill_rule          | 36 |
| x\$_l_gcvl_arc_mode           | 40 |
| x\$_l_gcvl_tile               | 44 |
| x\$_l_gcvl_stipple            | 48 |
| x\$_l_gcvl_ts_x_origin        | 52 |
| x\$_l_gcvl_ts_y_origin        | 56 |
| x\$_l_gcvl_font               | 60 |
| x\$_l_gcvl_subwindow_mode     | 64 |
| x\$_l_gcvl_graphics_exposures | 68 |
| x\$_l_gcvl_clip_x_origin      | 72 |
| x\$_l_gcvl_clip_y_origin      | 76 |
| x\$_l_gcvl_clip_mask          | 80 |
| x\$_l_gcvl_dash_offset        | 84 |
| x\$_b_gcvl_dashes             |    |

## Xlib Data Structures and Error Codes

### 3.26 The GC Values Data Structure

Table 3–26 describes the members of the GC values data structure.

**Table 3–26 GC Values Data Structure Members**

| Member Name          | Contents                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|----------------------|----------------------|--------------------|------------|-------------|--------------------|-----------------|-------------|-----|---------------------|-------------------|-------------|-----|------------|-------------|-----------|------------|------------|-----------------------|--------------|-------------------|---------------|---------|-------------------|----------------|----------------------|---------|--------------------|------------------|-------------|----------------------|------------|---|
| XSL_GCVL_FUNCTION    | <p>Defines how the server computes pixel values when the client updates a section of the screen. The following lists available functions:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th style="text-align: left;">Constant</th> <th style="text-align: left;">Description</th> </tr> </thead> <tbody> <tr><td>XSC_GX_CLEAR</td><td>0</td></tr> <tr><td>xSc_GX_AND</td><td>src AND dst</td></tr> <tr><td>XSC_GX_AND_REVERSE</td><td>src AND NOT dst</td></tr> <tr><td>XSC_GX_COPY</td><td>src</td></tr> <tr><td>XSC_GX_AND_INVERTED</td><td>(NOT src) AND dst</td></tr> <tr><td>XSC_GX_NOOP</td><td>dst</td></tr> <tr><td>XSC_GX_XOR</td><td>src XOR dst</td></tr> <tr><td>XSC_GX_OR</td><td>src OR dst</td></tr> <tr><td>XSC_GX_NOR</td><td>(NOT src) AND NOT dst</td></tr> <tr><td>XSC_GX_EQUIV</td><td>(NOT src) XOR dst</td></tr> <tr><td>XSC_GX_INVERT</td><td>NOT dst</td></tr> <tr><td>XSC_GX_OR_REVERSE</td><td>src OR NOT dst</td></tr> <tr><td>XSC_GX_COPY_INVERTED</td><td>NOT src</td></tr> <tr><td>XSC_GX_OR_INVERTED</td><td>(NOT src) OR dst</td></tr> <tr><td>XSC_GX_NAND</td><td>(NOT src) OR NOT dst</td></tr> <tr><td>XSC_GX_SET</td><td>1</td></tr> </tbody> </table> | Constant         | Description          | XSC_GX_CLEAR         | 0                  | xSc_GX_AND | src AND dst | XSC_GX_AND_REVERSE | src AND NOT dst | XSC_GX_COPY | src | XSC_GX_AND_INVERTED | (NOT src) AND dst | XSC_GX_NOOP | dst | XSC_GX_XOR | src XOR dst | XSC_GX_OR | src OR dst | XSC_GX_NOR | (NOT src) AND NOT dst | XSC_GX_EQUIV | (NOT src) XOR dst | XSC_GX_INVERT | NOT dst | XSC_GX_OR_REVERSE | src OR NOT dst | XSC_GX_COPY_INVERTED | NOT src | XSC_GX_OR_INVERTED | (NOT src) OR dst | XSC_GX_NAND | (NOT src) OR NOT dst | XSC_GX_SET | 1 |
| Constant             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_CLEAR         | 0                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| xSc_GX_AND           | src AND dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_AND_REVERSE   | src AND NOT dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_COPY          | src                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_AND_INVERTED  | (NOT src) AND dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_NOOP          | dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_XOR           | src XOR dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_OR            | src OR dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_NOR           | (NOT src) AND NOT dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_EQUIV         | (NOT src) XOR dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_INVERT        | NOT dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_OR_REVERSE    | src OR NOT dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_COPY_INVERTED | NOT src                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_OR_INVERTED   | (NOT src) OR dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_NAND          | (NOT src) OR NOT dst                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSC_GX_SET           | 1                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSL_GCVL_PLANE_MASK  | Specifies the planes on which the server performs the bitwise computation of pixels, defined by XSL_GCVL_FUNCTION.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSL_GCVL_FOREGROUND  | Specifies an index to a color map for foreground color.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSL_GCVL_BACKGROUND  | Specifies an index to a color map for background color.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSL_GCVL_LINE_WIDTH  | Defines the width of a line in pixels.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSL_GCVL_LINE_STYLE  | <p>Defines which sections of the line the server draws. The following line styles and the constants that specify them are available:</p> <table style="width: 100%;"> <tr> <td style="width: 50%;">xSc_line_solid</td> <td style="width: 50%;">xSc_line_off_on_dash</td> </tr> <tr> <td>xSc_line_double_dash</td> <td></td> </tr> </table>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         | xSc_line_solid   | xSc_line_off_on_dash | xSc_line_double_dash |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| xSc_line_solid       | xSc_line_off_on_dash                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| xSc_line_double_dash |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| XSL_GCVL_CAP_STYLE   | <p>Defines how the server draws the endpoints of a path. The following cap styles and the constants that specify them are available:</p> <table style="width: 100%;"> <tr> <td style="width: 50%;">xSc_cap_not_last</td> <td style="width: 50%;">xSc_cap_round</td> </tr> <tr> <td>xSc_cap_but</td> <td>xSc_cap_projecting</td> </tr> </table>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | xSc_cap_not_last | xSc_cap_round        | xSc_cap_but          | xSc_cap_projecting |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| xSc_cap_not_last     | xSc_cap_round                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |
| xSc_cap_but          | xSc_cap_projecting                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |                  |                      |                      |                    |            |             |                    |                 |             |     |                     |                   |             |     |            |             |           |            |            |                       |              |                   |               |         |                   |                |                      |         |                    |                  |             |                      |            |   |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.26 The GC Values Data Structure

**Table 3–26 (Cont.) GC Values Data Structure Members**

| Member Name                 | Contents                                                                                                                                                                                                                                                                                                |
|-----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_GCVL_JOIN_STYLE         | Defines how the server draws corners for wide lines. The following join styles and the constants that specify them are available:<br>xSc_join_mitre                      xSc_join_bevel<br>xSc_join_round                                                                                               |
| XSL_GCVL_FILL_STYLE         | Specifies the contents of the source for line, text, and fill operations. The following fill styles and the constants that specify them are available:<br>xSc_fill_solid                      xSc_fill_opaque_stippled<br>xSc_fill_tiled                      xSc_fill_stippled                         |
| XSL_GCVL_FILL_RULE          | Defines what pixels the server draws along a path when a polygon is filled. The two available choices are xSc_even_odd_rule and xSc_winding_rule.                                                                                                                                                       |
| XSL_GCVL_ARC_MODE           | Controls how the server fills an arc. The available choices are specified by the constants xSc_arc_pie_slice and xSc_arc_chord.                                                                                                                                                                         |
| XSL_GCVL_TILE               | Specifies the pixmap the server uses for tiling operations.                                                                                                                                                                                                                                             |
| XSL_GCVL_STIPPLE            | Specifies the pixmap the server uses for stipple operations.                                                                                                                                                                                                                                            |
| XSL_GCVL_TS_X_ORIGIN        | Defines the origin for tiling and stipple operations.                                                                                                                                                                                                                                                   |
| XSL_GCVL_TS_Y_ORIGIN        | Defines the origin for tiling and stipple operations.                                                                                                                                                                                                                                                   |
| XSL_GCVL_FONT               | Specifies the font that the server uses for text operations.                                                                                                                                                                                                                                            |
| XSL_GCVL_SUBWINDOW_MODE     | Specifies whether inferior windows clip superior windows. The constant xSc_clip_by_children specifies that all viewable input-output children clip both source and destination windows. The constant xSc_include_inferiors specifies that inferior windows clip neither source nor destination windows. |
| XSL_GCVL_GRAPHICS_EXPOSURES | Specifies whether the server informs the client when the contents of a window region are lost.                                                                                                                                                                                                          |
| XSL_GCVL_CLIP_X_ORIGIN      | Defines the x-coordinate of the clip origin.                                                                                                                                                                                                                                                            |
| XSL_GCVL_CLIP_Y_ORIGIN      | Defines the y-coordinate of the clip origin.                                                                                                                                                                                                                                                            |
| XSL_GCVL_CLIP_MASK          | Identifies the pixmap the server uses to restrict write operations to the destination that is drawable.                                                                                                                                                                                                 |
| XSL_GCVL_DASH_OFFSET        | Specifies the pixel within the dash length sequence, defined by XST_GCVL_DASHES.                                                                                                                                                                                                                        |
| XST_GCVL_DASHES             | Specifies the length, in number of pixels, of each dash.                                                                                                                                                                                                                                                |

#### 3.26.1 GC Mask

Table 3–27 lists the predefined values for the graphics context (GC) bit masks and their meaning.

**Table 3–27 GC Mask Bits**

| OpenVMS Predefined Bit Value | Meaning When Set             |
|------------------------------|------------------------------|
| XSM_GC_FUNCTION              | Change the function member   |
| XSM_GC_PLANE_MASK            | Change the plane mask member |

(continued on next page)



**Table 3–27 (Cont.) GC Mask Bits**

| OpenVMS Predefined Bit Value | Meaning When Set                                  |
|------------------------------|---------------------------------------------------|
| XSM_GC_FOREGROUND            | Change the foreground member                      |
| XSM_GC_BACKGROUND            | Change the background member                      |
| XSM_GC_LINE_WIDTH            | Change the line width member                      |
| XSM_GC_LINE_STYLE            | Change the line style member                      |
| XSM_GC_CAP_STYLE             | Change the cap, endpoint, style member            |
| XSM_GC_JOIN_STYLE            | Change the join style member                      |
| XSM_GC_FILL_STYLE            | Change the fill style member                      |
| XSM_GC_FILL_RULE             | Change the fill rule member                       |
| XSM_GC_ARC_MODE              | Change the arc fill mode                          |
| XSM_GC_TILE                  | Change the tile pixmap identifier                 |
| XSM_GC_STIPPLE               | Change the stipple pixmap identifier              |
| XSM_GC_TILE_STIP_X_ORIGIN    | Change the x-coordinate for a tile/stipple origin |
| XSM_TILE_STIP_Y_ORIGIN       | Change the y-coordinate for a tile/stipple origin |
| XSM_GC_FONT                  | Change the font identifier                        |
| XSM_GC_SUBWINDOW_MODE        | Change the subwindow mode member                  |
| XSM_GC_GRAPHICS_EXPOSURE     | Change the graphics exposure flag                 |
| XSM_GC_CLIP_X_ORIGIN         | Change the x-coordinate for a clip origin         |
| XSM_GC_CLIP_Y_ORIGIN         | Change the y-coordinate for a clip origin         |
| XSM_GC_CLIP_MASK             | Change clip mask pixmap identifier                |
| XSM_GC_DASH_OFFSET           | Change the dash offset member                     |
| XSM_GC_DASH_LIST             | Change the dash list member                       |

### 3.27 Graphics Expose Event Data Structure

Figure 3–27 illustrates the graphics expose event data structure (X\$GRAPH\_EXPOSE\_EVENT).

**Figure 3–27 Graphics Expose Event Data Structure**

|                      |    |
|----------------------|----|
| x\$l_geev_type       | 0  |
| x\$l_geev_serial     | 4  |
| x\$l_geev_send_event | 8  |
| x\$a_geev_display    | 12 |
| x\$l_geev_drawable   | 16 |
| x\$l_geev_x          | 20 |
| x\$l_geev_y          | 24 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.27 Graphics Expose Event Data Structure

|                      |    |
|----------------------|----|
| x\$l_geev_width      | 28 |
| x\$l_geev_height     | 32 |
| x\$l_geev_count      | 36 |
| x\$l_geev_major_code | 40 |
| x\$l_geev_minor_code | 44 |

Table 3–28 describes the members of the graphics expose event data structure.

**Table 3–28 Graphics Expose Event Data Structure Members**

| Member Name         | Contents                                                                                                                                                    |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_GEEV_TYPE       | Value defined by the constant xSc_graphics_expose.                                                                                                          |
| XSL_GEEV_SERIAL     | Number of the last request processed by the server.                                                                                                         |
| XSL_GEEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                                                                             |
| XSL_GEEV_DISPLAY    | Display on which the event occurred.                                                                                                                        |
| XSL_GEEV_DRAWABLE   | Window or pixmap reporting the event.                                                                                                                       |
| XSL_GEEV_X          | The x value of the coordinates that define the upper left corner of the exposed region. The coordinates are relative to the origin of the drawable.         |
| XSL_GEEV_Y          | The y value of the coordinates that define the upper left corner of the region that is exposed. The coordinates are relative to the origin of the drawable. |
| XSL_GEEV_WIDTH      | Width of the exposed region.                                                                                                                                |
| XSL_GEEV_HEIGHT     | Height of the exposed region.                                                                                                                               |
| XSL_GEEV_COUNT      | Number of exposure events that are to follow. If Xlib sets the count to zero, no more exposure events follow for this window.                               |
| XSL_GEEV_MAJOR_CODE | Indicates whether the graphics request was a copy area or copy plane.                                                                                       |
| XSL_GEEV_MINOR_CODE | The value zero. Reserved for use by extensions.                                                                                                             |

### 3.28 Gravity Event Data Structure

Figure 3–28 illustrates the gravity event data structure (X\$GRAVITY\_EVENT).

**Figure 3–28 Gravity Event Data Structure**

|                  |   |
|------------------|---|
| x\$l_gvev_type   | 0 |
| x\$l_gvev_serial | 4 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.28 Gravity Event Data Structure

|                      |    |
|----------------------|----|
| x\$I_gvev_send_event | 8  |
| x\$a_gvev_display    | 12 |
| x\$I_gvev_event      | 16 |
| x\$I_gvev_window     | 20 |
| x\$I_gvev_x          | 24 |
| x\$I_gvev_y          | 28 |

Table 3–29 describes the members of the gravity event data structure.

**Table 3–29 Gravity Event Data Structure Members**

| Member Name         | Contents                                                                                                                               |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| XSL_GVEV_TYPE       | Value defined by the x\$C_gravity_notify constant.                                                                                     |
| XSL_GVEV_SERIAL     | Number of the last request processed by the server.                                                                                    |
| XSL_GVEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                                                        |
| XSA_GVEV_DISPLAY    | Display on which the event occurred.                                                                                                   |
| XSL_GVEV_EVENT      | Event window.                                                                                                                          |
| XSL_GVEV_WINDOW     | Child window that has moved.                                                                                                           |
| XSL_GVEV_X          | The x value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window. |
| XSL_GVEV_Y          | The y value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window. |

### 3.29 Host Address Data Structure

Figure 3–29 illustrates the host address data structure (X\$HOST\_ADDRESS).

**Figure 3–29 Host Address Data Structure**

|                   |   |
|-------------------|---|
| x\$I_host_family  | 0 |
| x\$I_host_length  | 4 |
| x\$a_host_address | 8 |

## Xlib Data Structures and Error Codes

### 3.29 Host Address Data Structure

Table 3–30 describes the members of the host address data structure.

**Table 3–30 Host Address Data Structure Members**

| Member Name      | Contents                                                                                                       |
|------------------|----------------------------------------------------------------------------------------------------------------|
| XSL_HOST_FAMILY  | Specifies which protocol address family to use. The constant XSC_FAMILY_DECNET identifies the DECnet protocol. |
| XSL_HOST_LENGTH  | The length of the address, in bytes.                                                                           |
| XSA_HOST_ADDRESS | A pointer to host address.                                                                                     |

### 3.30 Icon Size Data Structure

Figure 3–30 illustrates the icon size data structure (X\$ICON\_SIZE).

**Figure 3–30 Icon Size Data Structure**

|                      |    |
|----------------------|----|
| x\$I_icsz_min_width  | 0  |
| x\$I_icsz_min_height | 4  |
| x\$I_icsz_max_width  | 8  |
| x\$I_icsz_max_height | 12 |
| x\$I_icsz_width_inc  | 16 |
| x\$I_icsz_height_inc | 20 |

Table 3–31 describes the members of the icon size data structure.

**Table 3–31 Icon Size Data Structure Members**

| Member Name                                | Contents                                                           |
|--------------------------------------------|--------------------------------------------------------------------|
| XSL_ICSZ_MIN_WIDTH,<br>XSL_ICSZ_MIN_HEIGHT | The minimum size for an icon window.                               |
| XSL_ICSZ_MAX_WIDTH,<br>XSL_ICSZ_MAX_HEIGHT | The maximum size for an icon window.                               |
| XSL_ICSZ_WIDTH_INC,<br>XSL_ICSZ_HEIGHT_INC | Specifies increments that can be added to a base width and height. |

### 3.31 Image Data Structure

Figure 3–31 illustrates the image data structure (X\$IMAGE).

**Figure 3–31 Image Data Structure**

|                            |    |
|----------------------------|----|
| x\$I_imag_width            | 0  |
| x\$I_imag_height           | 4  |
| x\$I_imag_xoffset          | 8  |
| x\$I_imag_format           | 12 |
| x\$a_imag_data             | 16 |
| x\$I_imag_byte_order       | 20 |
| x\$I_imag_bitmap_unit      | 24 |
| x\$I_imag_bitmap_bit_order | 28 |
| x\$I_imag_bitmap_pad       | 32 |
| x\$I_imag_depth            | 36 |
| x\$I_imag_bytes_per_line   | 40 |
| x\$I_imag_bits_per_pixel   | 44 |
| x\$I_imag_red_mask         | 48 |
| x\$I_imag_green_mask       | 52 |
| x\$I_imag_blue_mask        | 56 |
| x\$a_imag_obdata           | 60 |
| x\$a_imag_create_image     | 64 |
| x\$a_imag_destroy_image    | 68 |
| x\$a_imag_get_pixel        | 72 |
| x\$a_imag_put_pixel        | 76 |
| x\$a_imag_sub_image        | 80 |
| x\$a_imag_add_pixel        | 84 |

## Xlib Data Structures and Error Codes

### 3.31 Image Data Structure

Table 3–32 describes the members of the image data structure.

**Table 3–32 Image Data Structure Members**

| Member Name               | Contents                                                                                                                                                                                       |
|---------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_IMAG_WIDTH            | Specifies the width of the image.                                                                                                                                                              |
| XSL_IMAG_HEIGHT           | Specifies the height of the image.                                                                                                                                                             |
| XSL_IMAG_OFFSET           | Specifies the number of pixels offset in the x direction.                                                                                                                                      |
| XSL_IMAG_FORMAT           | Specifies whether the data is stored in XYPixmap or ZPixmap format. The following flags facilitate specifying data format:<br>xSc_xy_bitmap                      xSc_z_pixmap<br>xSc_xy_pixmap |
| XSA_IMAG_DATA             | The address of the image data.                                                                                                                                                                 |
| XSL_IMAG_BYTE_ORDER       | Indicates whether the least significant or the most significant byte is first.                                                                                                                 |
| XSL_IMAG_BITMAP_UNIT      | Specifies whether the bitmap is organized in units of 8, 16, or 32 bits.                                                                                                                       |
| XSL_IMAG_BITMAP_BIT_ORDER | Specifies whether the bitmap order is least or most significant.                                                                                                                               |
| XSL_IMAG_BITMAP_PAD       | Specifies whether padding in XY format or Z format should be done in units of 8, 16, or 32 bits.                                                                                               |
| XSL_IMAG_DEPTH            | The depth of the image.                                                                                                                                                                        |
| XSL_IMAG_BYTES_PER_LINE   | The bytes per line to be used as an accelerator.                                                                                                                                               |
| XSL_IMAG_BITS_PER_PIXEL   | Indicates for Z format the number of bits per pixel.                                                                                                                                           |
| XSL_IMAG_RED_MASK         | Specifies the red values for Z format.                                                                                                                                                         |
| XSL_IMAG_GREEN_MASK       | Specifies the green values for Z format.                                                                                                                                                       |
| XSL_IMAG_BLUE_MASK        | Specifies the blue values for Z format.                                                                                                                                                        |
| XSA_IMAG_OBDATA           | A structure that contains object routines.                                                                                                                                                     |
| XSA_IMAG_CREATE_IMAGE     | A CREATE IMAGE routine.                                                                                                                                                                        |
| XSA_IMAG_DESTROY_IMAGE    | A DESTROY IMAGE routine.                                                                                                                                                                       |
| XSA_IMAG_GET_PIXEL        | A GET PIXEL routine.                                                                                                                                                                           |
| XSA_IMAG_PUT_PIXEL        | A PUT IMAGE routine.                                                                                                                                                                           |
| XSA_IMAG_SUB_IMAGE        | A SUB IMAGE routine.                                                                                                                                                                           |
| XSA_IMAG_ADD_PIXEL        | An ADD PIXEL routine.                                                                                                                                                                          |

### 3.32 Key Event Data Structure

Figure 3–32 illustrates the key event data structure (XSKEY\_EVENT).

**Figure 3–32 Key Event Data Structure**

|                        |    |
|------------------------|----|
| x\$_l_kyev_type        | 0  |
| x\$_l_kyev_serial      | 4  |
| x\$_l_kyev_send_event  | 8  |
| x\$_a_kyev_display     | 12 |
| x\$_l_kyev_window      | 16 |
| x\$_l_kyev_root        | 20 |
| x\$_l_kyev_subwindow   | 24 |
| x\$_l_kyev_time        | 28 |
| x\$_l_kyev_x           | 32 |
| x\$_l_kyev_y           | 36 |
| x\$_l_kyev_x_root      | 40 |
| x\$_l_kyev_y_root      | 44 |
| x\$_l_kyev_state       | 48 |
| x\$_l_kyev_keycode     | 52 |
| x\$_l_kyev_same_screen | 56 |

Table 3–33 describes the members of the key event data structure.

**Table 3–33 Key Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_KYEV_TYPE       | Value defined by either the x\$C_key_press or the x\$C_key_release constant.    |
| XSL_KYEV_SERIAL     | Number of the last event processed by the server.                               |
| XSL_KYEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| XSA_KYEV_DISPLAY    | Display on which the event occurred.                                            |
| XSL_KYEV_WINDOW     | Event window.                                                                   |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.32 Key Event Data Structure

**Table 3–33 (Cont.) Key Event Data Structure Members**

| Member Name          | Contents                                                                                                                                                                                                                                                                                                                                                                              |            |           |              |           |           |           |           |           |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------|-----------|--------------|-----------|-----------|-----------|-----------|-----------|
| XSL_KYEV_ROOT        | Root window on which the event occurred.                                                                                                                                                                                                                                                                                                                                              |            |           |              |           |           |           |           |           |
| XSL_KYEV_SUBWINDOW   | Source window of the event.                                                                                                                                                                                                                                                                                                                                                           |            |           |              |           |           |           |           |           |
| XSL_KYEV_TIME        | Time in milliseconds at which the key event occurred.                                                                                                                                                                                                                                                                                                                                 |            |           |              |           |           |           |           |           |
| XSL_KYEV_X           | The x value of the pointer coordinates in the source window.                                                                                                                                                                                                                                                                                                                          |            |           |              |           |           |           |           |           |
| XSL_KYEV_Y           | The y value of the pointer coordinates in the source window.                                                                                                                                                                                                                                                                                                                          |            |           |              |           |           |           |           |           |
| XSL_KYEV_X_ROOT      | The x value of the pointer coordinates relative to the root window.                                                                                                                                                                                                                                                                                                                   |            |           |              |           |           |           |           |           |
| XSL_KYEV_Y_ROOT      | The y value of the pointer coordinates relative to the root window.                                                                                                                                                                                                                                                                                                                   |            |           |              |           |           |           |           |           |
| XSL_KYEV_STATE       | State of the key just prior to the key event. Xlib can set this member to the bitwise OR of the following states:<br><table style="margin-left: 2em; border: none;"> <tr> <td>x\$m_shift</td> <td>x\$m_lock</td> </tr> <tr> <td>x\$m_control</td> <td>x\$m_mod1</td> </tr> <tr> <td>x\$m_mod2</td> <td>x\$m_mod3</td> </tr> <tr> <td>x\$m_mod4</td> <td>x\$m_mod5</td> </tr> </table> | x\$m_shift | x\$m_lock | x\$m_control | x\$m_mod1 | x\$m_mod2 | x\$m_mod3 | x\$m_mod4 | x\$m_mod5 |
| x\$m_shift           | x\$m_lock                                                                                                                                                                                                                                                                                                                                                                             |            |           |              |           |           |           |           |           |
| x\$m_control         | x\$m_mod1                                                                                                                                                                                                                                                                                                                                                                             |            |           |              |           |           |           |           |           |
| x\$m_mod2            | x\$m_mod3                                                                                                                                                                                                                                                                                                                                                                             |            |           |              |           |           |           |           |           |
| x\$m_mod4            | x\$m_mod5                                                                                                                                                                                                                                                                                                                                                                             |            |           |              |           |           |           |           |           |
| XSL_KYEV_KEYCODE     | An arbitrary but unique representation of the key that generated the event.                                                                                                                                                                                                                                                                                                           |            |           |              |           |           |           |           |           |
| XSL_KYEV_SAME_SCREEN | Indicates whether the event window is on the same screen as the root window.                                                                                                                                                                                                                                                                                                          |            |           |              |           |           |           |           |           |

### 3.33 Keyboard Control Data Structure

Figure 3–33 illustrates the keyboard control data structure (X\$KEYBOARD\_CONTROL).

**Figure 3–33 Keyboard Control Data Structure**

|                             |    |
|-----------------------------|----|
| x\$I_kbdc_key_click_percent | 0  |
| x\$I_kbdc_bell_percent      | 4  |
| x\$I_kbdc_bell_pitch        | 8  |
| x\$I_kbdc_bell_duration     | 12 |
| x\$I_kbdc_led               | 16 |
| x\$I_kbdc_led_mode          | 20 |
| x\$I_kbdc_key               | 24 |
| x\$I_kbdc_auto_repeat_mode  | 28 |



Table 3–34 describes the members of the keyboard control data structure.

**Table 3–34 Keyboard Control Data Structure Members**

| Member Name                | Contents                               |
|----------------------------|----------------------------------------|
| XSL_KBDC_KEY_CLICK_PERCENT | Controls the volume for key clicks.    |
| XSL_KBDC_BELL_PERCENT      | Controls the base volume for the bell. |
| XSL_KBDC_BELL_PITCH        | Controls the pitch.                    |
| XSL_KBDC_BELL_DURATION     | Controls the duration.                 |
| XSL_KBDC_LED               | Changes the keyboard LED.              |
| XSL_KBDC_LED_MODE          | Changes the keyboard LED mode.         |
| XSL_KBDC_KEY               | Changes the keyboard autorepeat key.   |
| XSL_KBDC_AUTO_REPEAT_MODE  | Changes the keyboard autorepeat mode.  |

### 3.34 Keyboard State Data Structure

Figure 3–34 illustrates the keyboard state data structure (X\$KEYBOARD\_STATE).

**Figure 3–34 Keyboard State Data Structure**

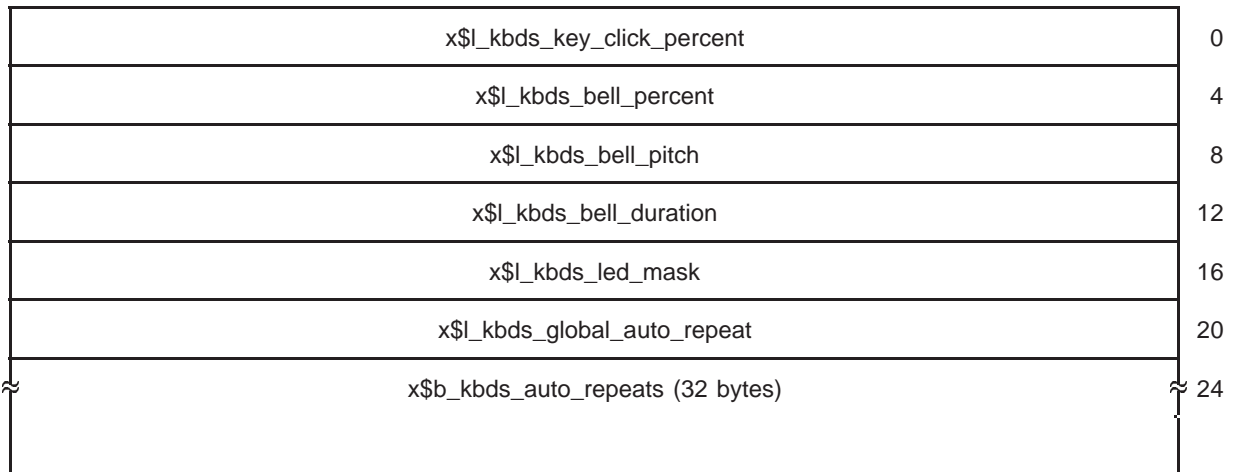


Table 3–35 describes the members of the keyboard state data structure.

**Table 3–35 Keyboard State Data Structure Members**

| Member Name                | Contents                      |
|----------------------------|-------------------------------|
| XSL_KBDS_KEY_CLICK_PERCENT | The key click percent value.  |
| XSL_KBDS_BELL_PERCENT      | The base volume for the bell. |
| XSL_KBDS_BELL_PITCH        | The bell pitch.               |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.34 Keyboard State Data Structure

**Table 3–35 (Cont.) Keyboard State Data Structure Members**

| Member Name                 | Contents                                                                                        |
|-----------------------------|-------------------------------------------------------------------------------------------------|
| XSL_KBDS_BELL_DURATION      | The bell duration.                                                                              |
| XSL_KBDS_LED_MASK           | The least significant bit corresponds to LED 1, and each one bit indicates an LED that is lit.  |
| XSL_KBDS_GLOBAL_AUTO_REPEAT | Global auto-repeat can be set either on or off.                                                 |
| XSB_KBDS_AUTO_REPEATS       | A bit vector where each one bit indicates that autorepeat is enabled for the corresponding key. |

### 3.35 Keymap Event Data Structure

Figure 3–35 illustrates the keymap event data structure (X\$KEYMAP\_EVENT).

**Figure 3–35 Keymap Event Data Structure**

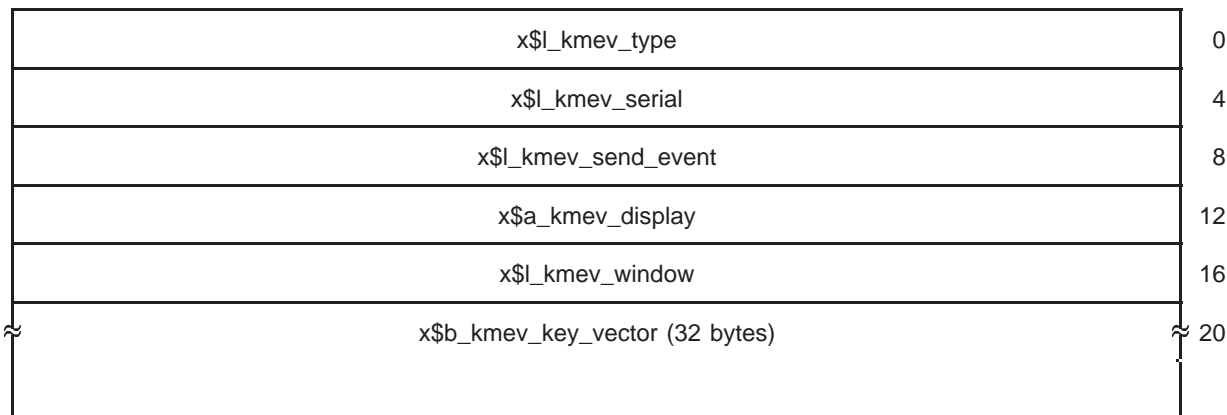


Table 3–36 describes the members of the keymap event data structure.

**Table 3–36 Keymap Event Data Structure Members**

| Member Name         | Contents                                                                                                  |
|---------------------|-----------------------------------------------------------------------------------------------------------|
| XSL_KMEV_TYPE       | Value defined by the <code>x\$C_keymap_notify</code> constant.                                            |
| XSL_KMEV_SERIAL     | Number of the last request processed by the server.                                                       |
| XSL_KMEV_SEND_EVENT | Value defined by the constant <code>true</code> if the event came from a <code>SEND EVENT</code> request. |
| XSA_KMEV_DISPLAY    | Display on which the event occurred.                                                                      |
| XSL_KMEV_WINDOW     | Event window.                                                                                             |
| XSB_KMEV_KEY_VECTOR | Bit vector of the keyboard.                                                                               |

### 3.36 Map Event Data Structure

Figure 3–36 illustrates the map event data structure (X\$MAP\_EVENT).

**Figure 3–36 Map Event Data Structure**

|                             |    |
|-----------------------------|----|
| x\$I_mpev_type              | 0  |
| x\$I_mpev_serial            | 4  |
| x\$I_mpev_send_event        | 8  |
| x\$a_mpev_display           | 12 |
| x\$I_mpev_event             | 16 |
| x\$I_mpev_window            | 20 |
| x\$I_mpev_override_redirect | 24 |

Table 3–37 describes the members of the map event data structure.

**Table 3–37 Map Event Data Structure Members**

| Member Name                | Contents                                                                                                                                                                                    |
|----------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_MPEV_TYPE              | Value defined by the xSc_map_notify constant.                                                                                                                                               |
| XSL_MPEV_SERIAL            | Number of the last request processed by the server.                                                                                                                                         |
| XSL_MPEV_SEND_EVENT        | Value defined by the constant true if the event came from a SEND EVENT request.                                                                                                             |
| XSA_MPEV_DISPLAY           | Display on which the event occurred.                                                                                                                                                        |
| XSL_MPEV_EVENT             | Event window.                                                                                                                                                                               |
| XSL_MPEV_WINDOW            | Window that has been mapped.                                                                                                                                                                |
| XSL_MPEV_OVERRIDE_REDIRECT | If the value of this member is defined by the constant true, the window manager should disregard requests to map the window. When true, it overrides a substructure redirect on the parent. |

### 3.37 Map Request Event Data Structure

Figure 3–37 illustrates the map request event data structure (X\$MAP\_REQUEST\_EVENT).

## Xlib Data Structures and Error Codes

### 3.37 Map Request Event Data Structure

Figure 3–37 Map Request Event Data Structure

|                      |    |
|----------------------|----|
| x\$I_mrev_type       | 0  |
| x\$I_mrev_serial     | 4  |
| x\$I_mrev_send_event | 8  |
| x\$a_mrev_display    | 12 |
| x\$I_mrev_parent     | 16 |
| x\$I_mrev_window     | 20 |

Table 3–38 describes the members of the map request event data structure.

Table 3–38 Map Request Event Data Structure Members

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_MREV_TYPE       | Value defined by the x\$c_mapping_notify constant.                              |
| XSL_MREV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_MREV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| X\$a_MREV_DISPLAY   | Display on which the event occurred.                                            |
| XSL_MREV_PARENT     | Parent window.                                                                  |
| XSL_MREV_WINDOW     | Window to be mapped.                                                            |

### 3.38 Mapping Event Data Structure

Figure 3–38 illustrates the mapping event data structure (X\$MAPPING\_EVENT).

Figure 3–38 Mapping Event Data Structure

|                      |    |
|----------------------|----|
| x\$I_mppg_type       | 0  |
| x\$I_mppg_serial     | 4  |
| x\$I_mppg_send_event | 8  |
| x\$a_mppg_display    | 12 |
| x\$I_mppg_window     | 16 |
| x\$I_mppg_request    | 20 |

(continued on next page)

|                         |    |
|-------------------------|----|
| x\$l_mppg_first_keycode | 24 |
| x\$l_mppg_count         | 28 |

Table 3–39 describes the members of the mapping event data structure.

**Table 3–39 Mapping Event Data Structure Members**

| Member Name            | Contents                                                                                                                                                                       |
|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_MPPG_TYPE          | Value defined by the x\$C_mapping_notify constant.                                                                                                                             |
| XSL_MPPG_SERIAL        | Number of the last request processed by the server.                                                                                                                            |
| XSL_MPPG_SEND_EVENT    | Value defined by the constant true if the event came from a SEND EVENT request.                                                                                                |
| XSA_MPPG_DISPLAY       | Display on which the event occurred.                                                                                                                                           |
| XSL_MPPG_WINDOW        | Unused member.                                                                                                                                                                 |
| XSL_MPPG_REQUEST       | The type of mapping change being reported. Possible values are indicated by the following constants:<br>x\$C_mapping_modifier<br>x\$C_mapping_keyboard<br>x\$C_mapping_pointer |
| XSL_MPPG_FIRST_KEYCODE | First number of the range of altered keys, set only if the request member has a value specified by the constant x\$C_mapping_keyboard.                                         |
| XSL_MPPG_COUNT         | Last number of the range of altered keys, set only if the request member has a value specified by the constant x\$C_mapping_keyboard.                                          |

### 3.39 Modifier Keymap Data Structure

Figure 3–39 illustrates the modifier keymap data structure (X\$MODIFIER\_KEYMAP).

**Figure 3–39 Modifier Keymap Data Structure**

|                          |   |
|--------------------------|---|
| x\$l_mdkey_max_keypermod | 0 |
| x\$a_mdkey_modifiermap   | 4 |

## Xlib Data Structures and Error Codes

### 3.39 Modifier Keymap Data Structure

Table 3–40 describes the members of the modifier keymap data structure.

**Table 3–40 Modifier Keymap Data Structure Members**

| Member Name            | Contents                                                                                                                    |
|------------------------|-----------------------------------------------------------------------------------------------------------------------------|
| XSL_MDKY_MAX_KEYPERMOD | Maximum number of keys per modifier of the server.                                                                          |
| XSA_MMDKY_MODIFIERMAP  | Array of modifiers. The array size is calculated by multiplying the value set in the XSL_MMDKY_MODIFIERMAP member by eight. |

### 3.40 Motion Event Data Structure

Figure 3–40 illustrates the motion event data structure (XSMOTION\_EVENT).

**Figure 3–40 Motion Event Data Structure**

|   |                         |    |
|---|-------------------------|----|
|   | x\$_mteev_type          | 0  |
|   | x\$_mteev_serial        | 4  |
|   | x\$_mteev_send_event    | 8  |
|   | x\$_mteev_display       | 12 |
|   | x\$_mteev_window        | 16 |
|   | x\$_mteev_root          | 20 |
|   | x\$_mteev_subwindow     | 24 |
|   | x\$_mteev_time          | 28 |
|   | x\$_mteev_x             | 32 |
|   | x\$_mteev_y             | 36 |
|   | x\$_mteev_x_root        | 40 |
|   | x\$_mteev_y_root        | 44 |
|   | x\$_mteev_state         | 48 |
| ↔ | x\$_mteev_same_screen   | 52 |
|   | x\$_mteev_is_hint       | 52 |
|   | x\$_mteev_same_screen : |    |

## Xlib Data Structures and Error Codes

### 3.40 Motion Event Data Structure

Table 3–41 describes the members of the motion event data structure.

**Table 3–41 Motion Event Data Structure Members**

| Member Name          | Contents                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_MTEV_TYPE        | Type of event reported. The member can have only the value specified by the constant <code>xSc_motion_notify</code> .                                                                                                                                                                                                                                                                                                                                                                                                        |
| XSL_MTEV_SERIAL      | Number of the last request processed by the server.                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| XSL_MTEV_SEND_EVENT  | Value defined by the constant <code>true</code> if the event came from a <code>SEND_EVENT</code> request.                                                                                                                                                                                                                                                                                                                                                                                                                    |
| XSA_MTEV_DISPLAY     | Display on which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| XSL_MTEV_WINDOW      | Event window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| XSL_MTEV_ROOT        | Root window in which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| XSL_MTEV_SUBWINDOW   | Source window in which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| XSL_MTEV_TIME        | Time in milliseconds at which the event occurred.                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
| XSL_MTEV_X           | The x value of the pointer coordinates in the source window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| XSL_MTEV_Y           | The y value of the pointer coordinates in the source window.                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| XSL_MTEV_X_ROOT      | The x value of the pointer coordinates relative to the root window.                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| XSL_MTEV_Y_ROOT      | The y value of the pointer coordinates relative to the root window.                                                                                                                                                                                                                                                                                                                                                                                                                                                          |
| XSL_MTEV_STATE       | State of the button just prior to the event. Xlib can set this member to the bitwise OR of one or more of the following masks:<br><div style="margin-left: 2em;"> <code>x\$m_button1</code>                <code>x\$m_mod1</code><br/> <code>x\$m_button2</code>                <code>x\$m_mod2</code><br/> <code>x\$m_button3</code>                <code>x\$m_mod3</code><br/> <code>x\$m_button4</code>                <code>x\$m_mod4</code><br/> <code>x\$m_button5</code>                <code>x\$m_mod5</code> </div> |
| XSB_MTEV_IS_HINT     | Indicates that motion hints are active. No other events reported until pointer moves out of window.                                                                                                                                                                                                                                                                                                                                                                                                                          |
| XSL_MTEV_SAME_SCREEN | Indicates whether or not the event window is on the same screen as the root window.                                                                                                                                                                                                                                                                                                                                                                                                                                          |

### 3.41 No Expose Event Data Structure

Figure 3–41 illustrates the no expose event data structure (`X$NO_EXPOSE_EVENT`).

**Figure 3–41 No Expose Event Data Structure**

|                                   |    |
|-----------------------------------|----|
| <code>x\$l_neev_type</code>       | 0  |
| <code>x\$l_neev_serial</code>     | 4  |
| <code>x\$l_neev_send_event</code> | 8  |
| <code>x\$a_neev_display</code>    | 12 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.41 No Expose Event Data Structure

|                      |    |
|----------------------|----|
| x\$l_neev_drawable   | 16 |
| x\$l_neev_major_code | 20 |
| x\$l_neev_minor_code | 24 |

Table 3–42 describes the members of the no expose event data structure.

**Table 3–42 No Expose Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_NEEV_TYPE       | Value defined by the constant x\$C_no_expose.                                   |
| XSL_NEEV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_NEEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| XSA_NEEV_DISPLAY    | Display on which the event occurred.                                            |
| XSL_NEEV_DRAWABLE   | Window or pixmap reporting the event.                                           |
| XSL_NEEV_MAJOR_CODE | Indicates whether the graphics request was a copy area or a copy plane.         |
| XSL_NEEV_MINOR_CODE | The value zero. Reserved for use by extensions.                                 |

### 3.42 Pixmap Format Values Data Structure

Figure 3–42 illustrates the pixmap format values data structure (X\$PIXMAP\_FORMAT\_VALUES).

**Figure 3–42 Pixmap Format Values Data Structure**

|                         |   |
|-------------------------|---|
| x\$l_pfv_depth          | 0 |
| x\$l_pfv_bits_per_pixel | 4 |
| x\$l_pfv_scanline_pad   | 8 |

Table 3–43 describes the members of the pixmap format values data structure.

**Table 3–43 Pixmap Format Values Data Structure Members**

| Member Name            | Contents                                                               |
|------------------------|------------------------------------------------------------------------|
| XSL_PFV_DEPTH          | Depth of screen                                                        |
| XSL_PFV_BITS_PER_PIXEL | Number of bits used to hold each pixel                                 |
| XSL_PFV_SCANLINE_PAD   | Specifies whether padding should be done in units of 8, 16, or 32 bits |



### 3.43 Point Data Structure

Figure 3–43 illustrates the point data structure (X\$POINT).

**Figure 3–43 Point Data Structure**



Table 3–44 describes the members of the point data structure.

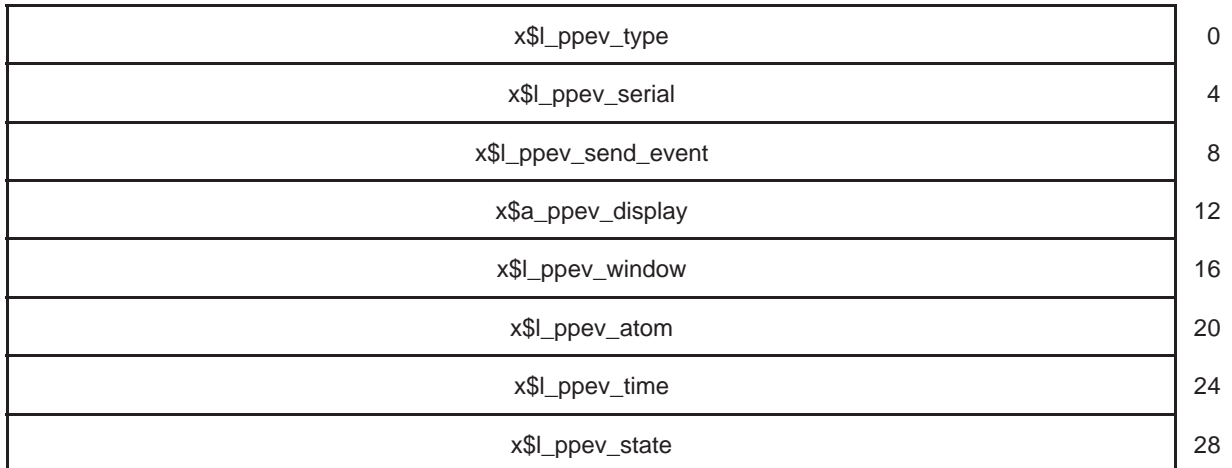
**Table 3–44 Point Data Structure Members**

| Member Name | Contents                                         |
|-------------|--------------------------------------------------|
| XSW_GPNT_X  | Defines the x value of the coordinate of a point |
| XSW_GPNT_Y  | Defines the y value of the coordinate of a point |

### 3.44 Property Event Data Structure

Figure 3–44 illustrates the property event data structure (X\$PROPERTY\_EVENT).

**Figure 3–44 Property Event Data Structure**



## Xlib Data Structures and Error Codes

### 3.44 Property Event Data Structure

Table 3–45 describes the members of the property event data structure.

**Table 3–45 Property Event Data Structure Members**

| Member Name         | Contents                                                                                                                                                    |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_PPEV_TYPE       | Value defined by the xSc_property_notify constant.                                                                                                          |
| XSL_PPEV_SERIAL     | Number of the last request processed by the server.                                                                                                         |
| XSL_PPEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                                                                             |
| XSA_PPEV_DISPLAY    | Display on which the event occurred.                                                                                                                        |
| XSL_PPEV_WINDOW     | Window whose property was changed.                                                                                                                          |
| XSL_PPEV_ATOM       | Identifies the property that was changed.                                                                                                                   |
| XSL_PPEV_TIME       | Server time that the property changed.                                                                                                                      |
| XSL_PPEV_STATE      | Value specified by the constant xSc_property_new_value if a client changes a property by calling either the CHANGE PROPERTY or the ROTATE PROPERTY routine. |

### 3.45 Rectangle Data Structure

Figure 3–45 illustrates the rectangle data structure (XSRECTANGLE).

**Figure 3–45 Rectangle Data Structure**

|                  |                 |   |
|------------------|-----------------|---|
| x\$w_grec_y      | x\$w_grec_x     | 0 |
| x\$w_grec_height | x\$w_grec_width | 4 |

Table 3–46 describes the members of the rectangle data structure.

**Table 3–46 Rectangle Data Structure Members**

| Member Name      | Contents                                    |
|------------------|---------------------------------------------|
| X\$W_GREC_X      | Defines the x value of the rectangle origin |
| X\$W_GREC_Y      | Defines the y value of the rectangle origin |
| X\$W_GREC_WIDTH  | Defines the width of the rectangle          |
| X\$W_GREC_HEIGHT | Defines the height of the rectangle         |

### 3.46 Reparent Event Data Structure

Figure 3–46 illustrates the reparent event data structure (X\$REPARENT\_EVENT).

Figure 3–46 Reparent Event Data Structure

|                             |    |
|-----------------------------|----|
| x\$I_rpev_type              | 0  |
| x\$I_rpev_serial            | 4  |
| x\$I_rpev_send_event        | 8  |
| x\$a_rpev_display           | 12 |
| x\$I_rpev_event             | 16 |
| x\$I_rpev_window            | 20 |
| x\$I_rpev_parent            | 24 |
| x\$I_rpev_x                 | 28 |
| x\$I_rpev_y                 | 32 |
| x\$I_rpev_override_redirect | 36 |

Table 3–47 describes the members of the reparent event data structure.

Table 3–47 Reparent Event Data Structure Members

| Member Name         | Contents                                                                                                                               |
|---------------------|----------------------------------------------------------------------------------------------------------------------------------------|
| XSL_RPEV_TYPE       | Value defined by the x\$C_reparent_notify constant.                                                                                    |
| XSL_RPEV_SERIAL     | Number of the last request processed by the server.                                                                                    |
| XSL_RPEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND_EVENT request.                                                        |
| XSA_RPEV_DISPLAY    | Display on which the event occurred.                                                                                                   |
| XSL_RPEV_EVENT      | Event window.                                                                                                                          |
| XSL_RPEV_WINDOW     | Window reparented.                                                                                                                     |
| XSL_RPEV_PARENT     | New parent of the window.                                                                                                              |
| XSL_RPEV_X          | The x value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window. |
| XSL_RPEV_Y          | The y value of the coordinates that define the upper left corner of the window relative to the upper left corner of the parent window. |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.46 Reparent Event Data Structure

**Table 3–47 (Cont.) Reparent Event Data Structure Members**

| Member Name                | Contents                                                                                                                                                                                   |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_RPEV_OVERRIDE_REDIRECT | If this member has a value defined by the constant true, the window manager ignores requests to reparent the window. When true, it overrides a substructure redirect on the parent window. |

### 3.47 Resize Request Event Data Structure

Figure 3–47 illustrates the resize request event data structure (X\$RESZ\_REQUEST\_EVENT).

**Figure 3–47 Resize Request Event Data Structure**

|                     |    |
|---------------------|----|
| x\$_rrev_type       | 0  |
| x\$_rrev_serial     | 4  |
| x\$_rrev_send_event | 8  |
| x\$a_rrev_display   | 12 |
| x\$_rrev_window     | 16 |
| x\$_rrev_width      | 20 |
| x\$_rrev_height     | 24 |

Table 3–48 describes the members of the resize request event data structure.

**Table 3–48 Resize Request Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_RREV_TYPE       | Value defined by the x\$C_resize_notify constant.                               |
| XSL_RREV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_RREV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| X\$a_RREV_DISPLAY   | Display on which the event occurred.                                            |
| XSL_RREV_WINDOW     | Window that client has requested to resize.                                     |
| XSL_RREV_WIDTH      | Inside width of window.                                                         |
| XSL_RREV_HEIGHT     | Inside height of window.                                                        |

### 3.48 Resource Manager Option Data Structure

Figure 3–48 illustrates the resource manager option data structure (X\$OPTIONS\_DESC\_REC).

**Figure 3–48 Resource Manager Option Data Structure**

|                         |    |
|-------------------------|----|
| x\$a_optd_option        | 0  |
| x\$a_optd_resource_name | 4  |
| x\$l_optd_arg_kind      | 8  |
| x\$a_optd_value         | 12 |

Table 3–49 describes the members of the resource manager option data structure.

**Table 3–49 Resource Manager Option Data Structure Members**

| Member Name             | Contents                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                     |                      |                     |                       |                         |                        |                      |                          |
|-------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------|----------------------|---------------------|-----------------------|-------------------------|------------------------|----------------------|--------------------------|
| XSA_OPTD_OPTION         | Option specification string in argv.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                     |                      |                     |                       |                         |                        |                      |                          |
| XSA_OPTD_RESOURCE_NAME  | Binding and resource name.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |                     |                      |                     |                       |                         |                        |                      |                          |
| XSL_OPTD_ARG_KIND       | Style of option. Xlib can set this member to one of the following constants:                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |                     |                      |                     |                       |                         |                        |                      |                          |
|                         | <table style="width: 100%; border: none;"> <tr> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_no_arg</td> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_res_arg</td> </tr> <tr> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_is_arg</td> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_skip_arg</td> </tr> <tr> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_sticky_arg</td> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_skip_line</td> </tr> <tr> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_sep_arg</td> <td style="padding: 2px 10px 2px 20px;">x\$rm_option_skip_n_args</td> </tr> </table> | x\$rm_option_no_arg | x\$rm_option_res_arg | x\$rm_option_is_arg | x\$rm_option_skip_arg | x\$rm_option_sticky_arg | x\$rm_option_skip_line | x\$rm_option_sep_arg | x\$rm_option_skip_n_args |
| x\$rm_option_no_arg     | x\$rm_option_res_arg                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |                     |                      |                     |                       |                         |                        |                      |                          |
| x\$rm_option_is_arg     | x\$rm_option_skip_arg                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |                     |                      |                     |                       |                         |                        |                      |                          |
| x\$rm_option_sticky_arg | x\$rm_option_skip_line                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |                     |                      |                     |                       |                         |                        |                      |                          |
| x\$rm_option_sep_arg    | x\$rm_option_skip_n_args                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |                     |                      |                     |                       |                         |                        |                      |                          |
| XSA_OPTD_VALUE          | Value if x\$rm_option_no_arg or x\$rm_option_skip_n_args used.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                          |                     |                      |                     |                       |                         |                        |                      |                          |

### 3.49 Resource Manager Value Data Structure

Figure 3–49 illustrates the resource manager value data structure (X\$RM\_VALUE).

**Figure 3–49 Resource Manager Value Data Structure**

|                |   |
|----------------|---|
| x\$l_rval_size | 0 |
| x\$a_rval_addr | 4 |

## Xlib Data Structures and Error Codes

### 3.49 Resource Manager Value Data Structure

Table 3–50 describes the members of the resource manager value data structure.

**Table 3–50 Resource Manager Value Data Structure Members**

| Member Name   | Contents                |
|---------------|-------------------------|
| XSL_RVAL_SIZE | Size of the resource    |
| XSA_RVAL_ADDR | Address of the resource |

### 3.50 Segment Data Structure

Figure 3–50 illustrates the segment data structure (XSSEGMENT).

**Figure 3–50 Segment Data Structure**

|              |              |   |
|--------------|--------------|---|
| x\$w_gseg_y1 | x\$w_gseg_x1 | 0 |
| x\$w_gseg_y2 | x\$w_gseg_x2 | 4 |

Table 3–51 describes the members of the segment data structure.

**Table 3–51 Segment Data Structure Members**

| Member Name | Contents                                                                       |
|-------------|--------------------------------------------------------------------------------|
| XSW_GSEG_X1 | The x value of the coordinate that specifies one endpoint of the segment       |
| XSW_GSEG_Y1 | The y value of the coordinate that specifies one endpoint of the segment       |
| XSW_GSEG_X2 | The x value of the coordinate that specifies the other endpoint of the segment |
| XSW_GSEG_Y2 | The y value of the coordinate that specifies the other endpoint of the segment |

### 3.51 Selection Clear Event Data Structure

Figure 3–51 illustrates the selection clear event data structure (X\$SELECT\_CLEAR\_EVENT).

## Xlib Data Structures and Error Codes

### 3.51 Selection Clear Event Data Structure

**Figure 3–51 Selection Clear Event Data Structure**

|                      |    |
|----------------------|----|
| x\$l_scev_type       | 0  |
| x\$l_scev_serial     | 4  |
| x\$l_scev_send_event | 8  |
| x\$a_scev_display    | 12 |
| x\$l_scev_window     | 16 |
| x\$l_scev_selection  | 20 |
| x\$l_scev_time       | 24 |

Table 3–52 describes the members of the selection clear event data structure.

**Table 3–52 Selection Clear Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_SCEV_TYPE       | Value defined by the x\$C_selection_clear constant.                             |
| XSL_SCEV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_SCEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |
| XSA_SCEV_DISPLAY    | Display on which the event occurred.                                            |
| XSL_SCEV_WINDOW     | Window losing ownership of the selection.                                       |
| XSL_SCEV_SELECTION  | Selection atom.                                                                 |
| XSL_SCEV_TIME       | Last time change recorded for the selection.                                    |

### 3.52 Selection Event Data Structure

Figure 3–52 illustrates the selection event data structure (X\$SELECTION\_EVENT).

**Figure 3–52 Selection Event Data Structure**

|                      |    |
|----------------------|----|
| x\$l_slev_type       | 0  |
| x\$l_slev_serial     | 4  |
| x\$l_slev_send_event | 8  |
| x\$a_slev_display    | 12 |
| x\$l_slev_requestor  | 16 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.52 Selection Event Data Structure

|                    |    |
|--------------------|----|
| x\$_slev_selection | 20 |
| x\$_slev_target    | 24 |
| x\$_slev_property  | 28 |
| x\$_slev_time      | 32 |

Table 3–53 describes the members of the selection event data structure.

**Table 3–53 Selection Event Data Structure Members**

| Member Name         | Contents                                                                                                    |
|---------------------|-------------------------------------------------------------------------------------------------------------|
| XSL_SLEV_TYPE       | Value defined by the x\$c_selection_notify constant.                                                        |
| XSL_SLEV_SERIAL     | Number of the last request processed by the server.                                                         |
| XSL_SLEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                             |
| X\$a_SLEV_DISPLAY   | Display on which the event occurred.                                                                        |
| XSL_SLEV_REQUESTOR  | Window that has requested the selection.                                                                    |
| XSL_SLEV_SELECTION  | Selection atom.                                                                                             |
| XSL_SLEV_TARGET     | Data type to which selection is converted.                                                                  |
| XSL_SLEV_PROPERTY   | Atom that specifies a property or the constant x\$c_none.                                                   |
| XSL_SLEV_TIME       | Timestamp, expressed in milliseconds, or the constant x\$c_current_time from the convert selection request. |

### 3.53 Selection Request Event Data Structure

Figure 3–53 illustrates the selection request event data structure (X\$SEL\_REQUEST\_EVENT).

**Figure 3–53 Selection Request Event Data Structure**

|                     |    |
|---------------------|----|
| x\$_srev_type       | 0  |
| x\$_srev_serial     | 4  |
| x\$_srev_send_event | 8  |
| x\$a_srev_display   | 12 |
| x\$_srev_owner      | 16 |
| x\$_srev_requestor  | 20 |
| x\$_srev_selection  | 24 |

(continued on next page)



## Xlib Data Structures and Error Codes

### 3.53 Selection Request Event Data Structure

|                    |    |
|--------------------|----|
| x\$l_srev_target   | 28 |
| x\$l_srev_property | 32 |
| x\$l_srev_time     | 36 |

Table 3–54 describes the members of the selection request event data structure.

**Table 3–54 Selection Request Event Data Structure Members**

| Member Name         | Contents                                                                                                     |
|---------------------|--------------------------------------------------------------------------------------------------------------|
| XSL_SREV_TYPE       | Value defined by the x\$sc_selection_request constant.                                                       |
| XSL_SREV_SERIAL     | Number of the last request processed by the server.                                                          |
| XSL_SREV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request.                              |
| XSA_SREV_DISPLAY    | Display on which the event occurred.                                                                         |
| XSL_SREV_OWNER      | Window that owns the selection.                                                                              |
| XSL_SREV_REQUESTOR  | Window that requests the selection.                                                                          |
| XSL_SREV_SELECTION  | Selection atom.                                                                                              |
| XSL_SREV_TARGET     | Data type that selection is converted to before being returned.                                              |
| XSL_SREV_PROPERTY   | Atom that specifies a property or the constant x\$sc_none.                                                   |
| XSL_SREV_TIME       | Timestamp, expressed in milliseconds, or the constant x\$sc_current_time from the convert selection request. |

### 3.54 Set Window Attributes Data Structure

Figure 3–54 illustrates the set window attributes data structure (X\$SET\_WIN\_ATTRIBUTES).

**Figure 3–54 Set Window Attributes Data Structure**

|                             |    |
|-----------------------------|----|
| x\$l_swda_background_pixmap | 0  |
| x\$l_swda_background_pixel  | 4  |
| x\$l_swda_border_pixmap     | 8  |
| x\$l_swda_border_pixel      | 12 |
| x\$l_swda_bit_gravity       | 16 |
| x\$l_swda_win_gravity       | 20 |
| x\$l_swda_backing_store     | 24 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.54 Set Window Attributes Data Structure

|                                |    |
|--------------------------------|----|
| x\$_swda_backing_planes        | 28 |
| x\$_swda_backing_pixel         | 32 |
| x\$_swda_save_under            | 36 |
| x\$_swda_event_mask            | 40 |
| x\$_swda_do_not_propagate_mask | 44 |
| x\$_swda_override_redirect     | 48 |
| x\$_swda_colormap              | 52 |
| x\$_swda_cursor                | 56 |

Table 3–55 describes the members of the set window attributes data structure.

**Table 3–55 Set Window Attributes Data Structure Members**

| Member Name                | Contents                                                                                                                                                                                               |
|----------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_SWDA_BACKGROUND_PIXMAP | Defines the window background of an input-output window. This member can assume one of three possible values: pixmap identifier, the constant xSc_none (default), or the constant xSc_parent_relative. |
| XSL_SWDA_BACKGROUND_PIXEL  | Specifying a value for the XSL_SWDA_BACKGROUND_PIXEL member causes the server to override the XSL_SWDA_BACKGROUND_PIXMAP member.                                                                       |
| XSL_SWDA_BORDER_PIXMAP     | Defines the window border of an input-output window. This member can be either XSC_COPY_FROM_PARENT or a pixmap identifier.                                                                            |
| XSL_SWDA_BORDER_PIXEL      | Specifying a value for XSL_SWDA_BORDER_PIXEL causes the server to override the XSL_SWDA_BORDER_PIXMAP member.                                                                                          |
| XSL_SWDA_BIT_GRAVITY       | Defines how window contents should be moved when an input-only or input-output window is resized.                                                                                                      |
| XSL_SWDA_WIN_GRAVITY       | Defines how the server should reposition the newly created input-only or input-output window when its parent window is resized.                                                                        |
| XSL_SWDA_BACKING_STORE     | Provides a hint to the server about how to manage obscured portions of the window.                                                                                                                     |
| XSL_SWDA_BACKING_PLANES    | Indicates (with bits set to one) which bit planes of the window hold dynamic data that must be preserved if the window obscures, or is obscured, by another window.                                    |
| XSL_SWDA_BACKING_PIXEL     | Defines what values to use in planes not specified by the XSL_SWDA_BACKING_PLANES member.                                                                                                              |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.54 Set Window Attributes Data Structure

**Table 3–55 (Cont.) Set Window Attributes Data Structure Members**

| Member Name                    | Contents                                                                                                                                                                     |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_SWDA_SAVE_UNDER            | Setting the XSL_SWDA_SAVE_UNDER member to true informs the server that the client requests the contents of the screen to be saved when an input-output window obscures them. |
| XSL_SWDA_EVENT_MASK            | Defines which types of events associated with an input-only or input-output window the server should report to the client.                                                   |
| XSL_SWDA_DO_NOT_PROPAGATE_MASK | Defines which kinds of events should not be propagated to ancestors.                                                                                                         |
| XSL_SWDA_OVERRIDE_REDIRECT     | Specifies whether calls to map and configure an input-only or input-output window should override a request by another client to redirect those calls.                       |
| XSL_SWDA_COLORMAP              | Specifies the color map, if any, that best reflects the colors of an input-output window.                                                                                    |
| XSL_SWDA_CURSOR                | Specifying a value for the cursor member causes the server to use a particular cursor when the pointer is in an input-only or input-output window.                           |

### 3.55 Size Hints Data Structure

Figure 3–55 illustrates the size hints data structure (X\$SIZE\_HINTS).

**Figure 3–55 Size Hints Data Structure**

|                      |    |
|----------------------|----|
| x\$I_szhn_flags      | 0  |
| x\$I_szhn_x          | 4  |
| x\$I_szhn_y          | 8  |
| x\$I_szhn_width      | 12 |
| x\$I_szhn_height     | 16 |
| x\$I_szhn_min_width  | 20 |
| x\$I_szhn_min_height | 24 |
| x\$I_szhn_max_width  | 28 |
| x\$I_szhn_max_height | 32 |
| x\$I_szhn_width_inc  | 36 |
| x\$I_szhn_height_inc | 40 |
| x\$I_szhn_mnas_x     | 44 |
| x\$I_szhn_mnas_y     | 48 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.55 Size Hints Data Structure

|                       |    |
|-----------------------|----|
| x\$l_szhn_mxas_x      | 52 |
| x\$l_szhn_mxas_y      | 56 |
| x\$l_szhn_base_width  | 60 |
| x\$l_szhn_base_height | 64 |
| x\$l_szhn_win_gravity | 68 |

Table 3–56 describes the members of the size hints data structure.

**Table 3–56 Size Hints Data Structure Members**

| Member Name          | Contents                                                                         |
|----------------------|----------------------------------------------------------------------------------|
| XSL_SZHN_FLAGS       | Defines the members to which the client assigns values                           |
| XSL_SZHN_X           | Specifies the x-coordinate that defines window position                          |
| XSL_SZHN_Y           | Specifies the y-coordinate that defines window position                          |
| XSL_SZHN_WIDTH       | Defines the width of the window                                                  |
| XSL_SZHN_HEIGHT      | Defines the height of the window                                                 |
| XSL_SZHN_MIN_WIDTH   | Specifies the minimum useful width of the window                                 |
| XSL_SZHN_MIN_HEIGHT  | Specifies the minimum useful height of the window                                |
| XSL_SZHN_MAX_WIDTH   | Specifies the maximum useful width of the window                                 |
| XSL_SZHN_MAX_HEIGHT  | Specifies the maximum useful height of the window                                |
| XSL_SZHN_WIDTH_INC   | Defines the increments by which the width of the window can be resized           |
| XSL_SZHN_HEIGHT_INC  | Defines the increments by which the height of the window can be resized          |
| XSL_SZHN_MNAS_X      | Specifies the minimum aspect ratio of the window with the XSL_SZHN_MNAS_Y member |
| XSL_SZHN_MNAS_Y      | Specifies the minimum aspect ratio of the window with the XSL_SZHN_MNAS_X member |
| XSL_SZHN_MXAS_X      | Specifies the maximum aspect ratio of the window with the XSL_SZHN_MXAS_Y member |
| XSL_SZHN_MXAS_Y      | Specifies the maximum aspect ratio of the window with the XSL_SZHN_MXAS_X member |
| XSL_SZHN_BASE_WIDTH  | Defines the desired width of the window                                          |
| XSL_SZHN_BASE_HEIGHT | Defines the desired height of the window                                         |
| XSL_SZHN_WIN_GRAVITY | Defines the region of the window that is to be retained when it is resized       |

### 3.56 Standard Color Map Data Structure

Figure 3–56 illustrates the standard color map data structure (X\$STANDARD\_COLORMAP).

## Xlib Data Structures and Error Codes

### 3.56 Standard Color Map Data Structure

Figure 3–56 Standard Color Map Data Structure

|                     |    |
|---------------------|----|
| x\$_scmp_colormap   | 0  |
| x\$_scmp_red_max    | 4  |
| x\$_scmp_red_mult   | 8  |
| x\$_scmp_green_max  | 12 |
| x\$_scmp_green_mult | 16 |
| x\$_scmp_blue_max   | 20 |
| x\$_scmp_blue_mult  | 24 |
| x\$_scmp_base_pixel | 28 |
| x\$_scmp_visual_id  | 32 |
| x\$_scmp_kill_id    | 36 |

Table 3–57 describes the members of the standard color map data structure.

**Table 3–57 Standard Color Map Data Structure Members**

| Member Name         | Contents                                                                                                                      |
|---------------------|-------------------------------------------------------------------------------------------------------------------------------|
| XSL_SCMP_COLORMAP   | A color map identifier returned by CREATE COLORMAP                                                                            |
| XSL_SCMP_RED_MAX    | The maximum number of red values                                                                                              |
| XSL_SCMP_RED_MULT   | Scale factor used to create a full pixel value                                                                                |
| XSL_SCMP_GREEN_MAX  | The maximum number of green values                                                                                            |
| XSL_SCMP_GREEN_MULT | Scale factor used to create a full pixel value                                                                                |
| XSL_SCMP_BLUE_MAX   | The maximum number of blue values                                                                                             |
| XSL_SCMP_BLUE_MULT  | Scale factor used to create a full pixel value                                                                                |
| XSL_SCMP_BASE_PIXEL | The base pixel value used to compose a full color index                                                                       |
| XSL_SCMP_VISUAL_ID  | ID number of the visual                                                                                                       |
| XSL_SCMP_KILL_ID    | Resource ID that indicates whether cells are to be released by freeing the color map ID or by calling the KILL CLIENT routine |

### 3.57 Text Item Data Structure

Figure 3–57 illustrates the text item data structure (XSTEXT\_ITEM).

## Xlib Data Structures and Error Codes

### 3.57 Text Item Data Structure

Figure 3–57 Text Item Data Structure

|                   |    |
|-------------------|----|
| x\$a_text_chars   | 0  |
| x\$l_text_n_chars | 4  |
| x\$l_text_delta   | 8  |
| x\$l_text_font    | 12 |

Table 3–58 describes the members of the text item data structure.

**Table 3–58 Text Item Data Structure Members**

| Member Name      | Contents                                        |
|------------------|-------------------------------------------------|
| XSA_TEXT_CHARS   | Address of a string of characters               |
| XSL_TEXT_N_CHARS | Number of characters in the string              |
| XSL_TEXT_DELTA   | Spacing before the start of the string          |
| XSL_TEXT_FONT    | Identifier of the font used to print the string |

### 3.58 Text Item 16 Data Structure

Figure 3–58 illustrates the text item 16 data structure (XSTEXT\_ITEM\_16).

Figure 3–58 Text Item 16 Data Structure

|                   |    |
|-------------------|----|
| x\$a_tx16_chars   | 0  |
| x\$l_tx16_n_chars | 4  |
| x\$l_tx16_delta   | 8  |
| x\$l_tx16_font    | 12 |

Table 3–59 describes the members of the text item 16 data structure.

**Table 3–59 Text Item 16 Data Structure Members**

| Member Name      | Contents                                                             |
|------------------|----------------------------------------------------------------------|
| XSA_TX16_CHARS   | Address of a string of characters stored in a char 2b data structure |
| XSL_TX16_N_CHARS | Number of characters in the string                                   |
| XSL_TX16_DELTA   | Spacing before the start of the string                               |
| XSL_TX16_FONT    | Identifier of the font used to print the string                      |

### 3.59 Text Property Data Structure

Figure 3–59 illustrates the text property data structure (X\$TEXT\_PROPERTY).

**Figure 3–59 Text Property Data Structure**

|   |                    |                    |     |
|---|--------------------|--------------------|-----|
| ↔ | x\$l_tntp_encoding | x\$b_tntp_value    | 0   |
| ↔ | x\$l_tntp_format   | x\$l_tntp_encoding | ∴ 4 |
| ↔ | x\$l_tntp_nitems   | x\$l_tntp_format   | ∴ 8 |
|   |                    | x\$l_tntp_nitems   | ∴   |

Table 3–60 describes the members of the text property data structure.

**Table 3–60 Text Property Data Structure Members**

| Member Name        | Contents                 |
|--------------------|--------------------------|
| XSB_TXTTP_VALUE    | Character string         |
| XSL_TXTTP_ENCODING | Type of encoding         |
| XSL_TXTTP_FORMAT   | Either 8, 16, or 32 bits |
| XSL_TXTTP_NITEMS   | Number of items in value |

### 3.60 Time Coordinate Data Structure

Figure 3–60 illustrates the time coordinate data structure (X\$TIME\_COORD).

## Xlib Data Structures and Error Codes

### 3.60 Time Coordinate Data Structure

Figure 3–60 Time Coordinate Data Structure

|                |             |   |
|----------------|-------------|---|
| x\$l_time_time |             | 0 |
| x\$w_time_y    | x\$w_time_x | 4 |

Table 3–61 describes the members of the time coordinate data structure.

**Table 3–61 Time Coordinate Data Structure Members**

| Member Name   | Contents                                                                                                     |
|---------------|--------------------------------------------------------------------------------------------------------------|
| XSL_TIMC_TIME | Set to the time, in milliseconds                                                                             |
| XSW_TIMC_X    | Set to the x-coordinate of the pointer cursor and is reported relative to the origin of the specified window |
| XSW_TIMC_Y    | Set to the y-coordinate of the pointer cursor and is reported relative to the origin of the specified window |

### 3.61 Unmap Event Data Structure

Figure 3–61 illustrates the unmap event data structure (XSUNMAP\_EVENT).

Figure 3–61 Unmap Event Data Structure

|                          |    |
|--------------------------|----|
| x\$l_umev_type           | 0  |
| x\$l_umev_serial         | 4  |
| x\$l_umev_send_event     | 8  |
| x\$a_umev_display        | 12 |
| x\$l_umev_event          | 16 |
| x\$l_umev_window         | 20 |
| x\$l_umev_from_configure | 24 |

Table 3–62 describes the members of the unmap event data structure.



**Table 3–62 Unmap Event Data Structure Members**

| Member Name             | Contents                                                                                                                                                                                                           |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_UMEV_TYPE           | Value defined by the xSc_unmap_notify constant.                                                                                                                                                                    |
| XSL_UMEV_SERIAL         | Number of the last request processed by the server.                                                                                                                                                                |
| XSL_UMEV_SEND_EVENT     | Value defined by the constant true if the event came from a SEND EVENT request.                                                                                                                                    |
| X\$a_UMEV_DISPLAY       | Display on which the event occurred.                                                                                                                                                                               |
| XSL_UMEV_EVENT          | Event window.                                                                                                                                                                                                      |
| XSL_UMEV_WINDOW         | Window unmapped.                                                                                                                                                                                                   |
| XSL_UMEV_FROM_CONFIGURE | If the value of this member is defined by the constant true, the event occurred as a result of resizing the parent window when the window itself has a window gravity specified by the constant xSc_unmap_gravity. |

### 3.62 Visibility Event Data Structure

Figure 3–62 illustrates the visibility event data structure (X\$VISIBILITY\_EVENT).

**Figure 3–62 Visibility Event Data Structure**

|                      |    |
|----------------------|----|
| x\$I_vsev_type       | 0  |
| x\$I_vsev_serial     | 4  |
| x\$I_vsev_send_event | 8  |
| x\$a_vsev_display    | 12 |
| x\$I_vsev_window     | 16 |
| x\$I_vsev_state      | 20 |

Table 3–63 describes the members of the visibility event data structure.

**Table 3–63 Visibility Event Data Structure Members**

| Member Name         | Contents                                                                        |
|---------------------|---------------------------------------------------------------------------------|
| XSL_VSEV_TYPE       | Value defined by the xSc_visibility_notify constant.                            |
| XSL_VSEV_SERIAL     | Number of the last request processed by the server.                             |
| XSL_VSEV_SEND_EVENT | Value defined by the constant true if the event came from a SEND EVENT request. |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.62 Visibility Event Data Structure

**Table 3–63 (Cont.) Visibility Event Data Structure Members**

| Member Name       | Contents                                                                                                                                                                                                    |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| X\$A_VSEV_DISPLAY | Display on which the event occurred.                                                                                                                                                                        |
| X\$S_VSEV_WINDOW  | Window whose visibility changed.                                                                                                                                                                            |
| X\$S_VSEV_STATE   | Determines the visibility state of the window. Xlib can set this member to one of the following constants:<br>x\$C_visibility_unobscured<br>x\$C_visibility_part_obscured<br>x\$C_visibility_fully_obscured |

### 3.63 Visual Info Data Structure

Figure 3–63 illustrates the visual info data structure (X\$VISUAL\_INFO).

**Figure 3–63 Visual Info Data Structure**

|                         |    |
|-------------------------|----|
| x\$a_visl_visual        | 0  |
| x\$l_visl_visual_id     | 4  |
| x\$l_visl_screen        | 8  |
| x\$l_visl_depth         | 12 |
| x\$l_visl_class         | 16 |
| x\$l_visl_red_mask      | 20 |
| x\$l_visl_green_mask    | 24 |
| x\$l_visl_blue_mask     | 28 |
| x\$l_visl_colormap_size | 32 |
| x\$l_visl_bits_per_rgb  | 36 |

Table 3–64 describes the members of the visual info data structure.

**Table 3–64 Visual Info Data Structure Members**

| Member Name         | Contents                                                             |
|---------------------|----------------------------------------------------------------------|
| X\$A_VISL_VISUAL    | A pointer to a visual data structure that is returned to the client. |
| X\$S_VISL_VISUAL_ID | The ID of the visual that is returned by the server.                 |
| X\$S_VISL_SCREEN    | The specified screen of the display.                                 |

(continued on next page)

**Table 3–64 (Cont.) Visual Info Data Structure Members**

| Member Name            | Contents                                                                                                                                  |
|------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_VISL_DEPTH         | The depth in planes of the screen.                                                                                                        |
| XSL_VISL_CLASS         | The class of the visual (x\$C_pseudo_color, x\$C_gray_scale, x\$C_direct_color, x\$C_true_color, x\$C_static_gray, or x\$C_static_color). |
| XSL_VISL_RED_MASK      | Definition of the red mask <sup>1</sup> .                                                                                                 |
| XSL_VISL_GREEN_MASK    | Definition of the green mask <sup>1</sup> .                                                                                               |
| XSL_VISL_BLUE_MASK     | Definition of the blue mask <sup>1</sup> .                                                                                                |
| XSL_VISL_COLORMAP_SIZE | Number of available color map entries.                                                                                                    |
| XSL_VISL_BITS_PER_RGB  | Number of bits that specifies the number of distinct red, green, and blue values.                                                         |

<sup>1</sup>The red mask, green mask, and blue mask are defined only for the direct color and true color visual types.

### 3.64 Window Attributes Data Structure

Figure 3–64 illustrates the window attributes data structure (X\$WINDOW\_ATTRIBUTES).

**Figure 3–64 Window Attributes Data Structure**

|                          |    |
|--------------------------|----|
| x\$I_wdat_x              | 0  |
| x\$I_wdat_y              | 4  |
| x\$I_wdat_width          | 8  |
| x\$I_wdat_height         | 12 |
| x\$I_wdat_border_width   | 16 |
| x\$I_wdat_depth          | 20 |
| x\$I_wdat_visual         | 24 |
| x\$I_wdat_root           | 28 |
| x\$I_wdat_class          | 32 |
| x\$I_wdat_bit_gravity    | 36 |
| x\$I_wdat_win_gravity    | 40 |
| x\$I_wdat_backing_store  | 44 |
| x\$I_wdat_backing_planes | 48 |
| x\$I_wdat_backing_pixel  | 52 |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.64 Window Attributes Data Structure

|                             |    |
|-----------------------------|----|
| x\$_wdat_save_under         | 56 |
| x\$_wdat_colormap           | 60 |
| x\$_wdat_map_installed      | 64 |
| x\$_wdat_map_state          | 68 |
| x\$_wdat_all_event_masks    | 72 |
| x\$_wdat_your_event_mask    | 76 |
| x\$_wdat_not_propagate_mask | 80 |
| x\$_wdat_override_redirect  | 84 |
| x\$_wdat_screen             | 88 |

Table 3–65 describes the members of the window attributes data structure.

**Table 3–65 Window Attributes Data Structure Members**

| Member Name           | Contents                                                                                                                   |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------|
| XSL_WDAT_X            | Specifies the x-coordinate of the upper left outside corner of the window relative to its parent window.                   |
| XSL_WDAT_Y            | Specifies the y-coordinate of the upper left outside corner of the window relative to its parent window.                   |
| XSL_WDAT_WIDTH        | Specifies the width of the window, excluding the window border, in pixels.                                                 |
| XSL_WDAT_HEIGHT       | Specifies the height of the window, excluding the window border, in pixels.                                                |
| XSL_WDAT_BORDER_WIDTH | Specifies the width of the window border in pixels.                                                                        |
| XSL_WDAT_DEPTH        | Specifies the bits per pixel of the window.                                                                                |
| XSL_WDAT_VISUAL       | The VISUAL structure associated with the window. The VISUAL structure specifies how displays should treat color resources. |
| XSL_WDAT_ROOT         | Identifies the screen with which the window is associated.                                                                 |
| XSL_WDAT_CLASS        | Specifies whether the window accepts input and output, or input only.                                                      |
| XSL_WDAT_BIT_GRAVITY  | Specifies how pixels should be moved when the window is resized.                                                           |
| XSL_WDAT_WIN_GRAVITY  | Specifies how the window should be repositioned when its parent window is resized.                                         |

(continued on next page)

## Xlib Data Structures and Error Codes

### 3.64 Window Attributes Data Structure

**Table 3–65 (Cont.) Window Attributes Data Structure Members**

| Member Name                    | Contents                                                                                                                                                                                                                                  |
|--------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_WDAT_BACKING_STORE         | Indicates whether the server should maintain a record of portions of a window that are obscured when the window is mapped. Xlib can set this member to one of the following constants:<br>xSc_when_mapped<br>xSc_always<br>xSc_not_useful |
| XSL_WDAT_BACKING_PLANES        | Indicates (with bits set to 1) which bit planes of the window hold dynamic data that must be preserved in backing stores and during save unders.                                                                                          |
| XSL_WDAT_BACKING_PIXEL         | Defines what values to use in planes not specified by XSL_WDAT_BACKING_PLANES.                                                                                                                                                            |
| XSL_WDAT_SAVE_UNDER            | Setting this member to true informs the server that the client would like the contents of the screen saved when the window obscures them. Saving the contents of obscured portions of the screen is not guaranteed.                       |
| XSL_WDAT_COLORMAP              | Specifies the color map, if any, that best reflects the colors of the window. The color map must have the same visual type as the window. If it does not, an error occurs.                                                                |
| XSL_WDAT_MAP_INSTALLED         | If set to true, indicates that the color map is currently installed and the window is being displayed in its correct colors.                                                                                                              |
| XSL_WDAT_MAP_STATE             | Indicates whether the window is mapped and viewable. Xlib can set this member to one of the following constants:<br>xSc_is_unmapped<br>xSc_is_unviewable<br>xSc_is_viewable                                                               |
| XSL_WDAT_ALL_EVENT_MASKS       | Indicates the set of events in which all applications have an interest. XSL_WDAT_ALL_EVENTS_MASK is the inclusive-OR of all event masks set for the window.                                                                               |
| XSL_WDAT_YOUR_EVENT_MASK       | Indicates the events about which the querying client is interested in receiving notice.                                                                                                                                                   |
| XSL_WDAT_DO_NOT_PROPAGATE_MASK | Defines which events should not be propagated to its ancestors when no application has the event type selected in the window.                                                                                                             |
| XSL_WDAT_OVERRIDE_REDIRECT     | Specifies whether requests to map and configure the window should override a request by another client to redirect those calls.                                                                                                           |
| XSL_WDAT_SCREEN                | Specifies the screen on which the window is mapped.                                                                                                                                                                                       |

## Xlib Data Structures and Error Codes

### 3.65 Window Changes Data Structure

### 3.65 Window Changes Data Structure

Figure 3–65 illustrates the window changes data structure (X\$WINDOW\_CHANGES).

Figure 3–65 Window Changes Data Structure

|                       |    |
|-----------------------|----|
| x\$_wchg_x            | 0  |
| x\$_wchg_y            | 4  |
| x\$_wchg_width        | 8  |
| x\$_wchg_height       | 12 |
| x\$_wchg_border_width | 16 |
| x\$_wchg_sibling      | 20 |
| x\$_wchg_stack_mode   | 24 |

Table 3–66 describes the members of the window changes data structure.

Table 3–66 Window Changes Data Structure Members

| Member Name           | Contents                                                                                                                                                                      |
|-----------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_WCHG_X            | Defines the x-coordinate of the new location of the window relative to the origin of its parent. The x and y coordinates specify the upper left outside corner of the window. |
| XSL_WCHG_Y            | Defines the y-coordinate of the new location of the window relative to the origin of its parent. The x and y coordinates specify the upper left outside corner of the window. |
| XSL_WCHG_WIDTH        | Defines the new width of the window, excluding the border.                                                                                                                    |
| XSL_WCHG_HEIGHT       | Defines the new height of the window, excluding the border.                                                                                                                   |
| XSL_WCHG_BORDER_WIDTH | Specifies the new window border in pixels.                                                                                                                                    |
| XSL_WCHG_SIBLING      | Specifies the sibling window for stacking order.                                                                                                                              |
| XSL_WCHG_STACK_MODE   | Defines how the window is restacked.                                                                                                                                          |

### 3.66 WM Hints Data Structure

Figure 3–66 illustrates the WM hints data structure (X\$WM\_HINTS).

**Figure 3–66 WM Hints Data Structure**

|                        |    |
|------------------------|----|
| x\$_hint_flags         | 0  |
| x\$_hint_input         | 4  |
| x\$_hint_initial_state | 8  |
| x\$_hint_icon_pixmap   | 12 |
| x\$_hint_icon_window   | 16 |
| x\$_hint_icon_x        | 20 |
| x\$_hint_icon_y        | 24 |
| x\$_hint_icon_mask     | 28 |
| x\$_hint_window_group  | 32 |

Table 3–67 describes the members of the WM hints data structure.

**Table 3–67 WM Hints Data Structure Members**

| Member Name            | Contents                                                                                                                                                                 |
|------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSL_HINT_FLAGS         | Specifies the members of the data structure that are defined.                                                                                                            |
| XSL_HINT_INPUT         | Indicates whether or not the client relies on the window manager for keyboard input.                                                                                     |
| XSL_HINT_INITIAL_STATE | Defines how the window should appear in its initial configuration. Possible initial states are as follows:<br>xSc_withdrawn_state<br>xSc_normal_state<br>xSc_ionic_state |
| XSL_HINT_ICON_PIXMAP   | Identifies the pixmap used to create the window icon.                                                                                                                    |
| XSL_HINT_ICON_WINDOW   | Specifies the window to be used as an icon.                                                                                                                              |
| XSL_HINT_ICON_X        | Specifies the initial x-coordinate of the icon position.                                                                                                                 |
| XSL_HINT_ICON_Y        | Specifies the initial y-coordinate of the icon position.                                                                                                                 |
| XSL_HINT_ICON_MASK     | Specifies the pixels of the icon pixmap used to create the icon.                                                                                                         |
| XSL_HINT_WINDOW_GROUP  | Specifies that a window belongs to a group of other windows.                                                                                                             |

### 3.67 Error Handling

Xlib routines can return the error codes described in Table 3–68. For more information about error handling, see the *X Window System* document.

## Xlib Data Structures and Error Codes

### 3.67 Error Handling

Table 3–68 Xlib Error Codes

| Error Code             | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
|------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSC_BAD_ACCESS         | <p>Possible causes are as follows:</p> <ul style="list-style-type: none"> <li>• An attempt to grab a key/button combination that has already been grabbed by another client</li> <li>• An attempt to free a color map entry that was not allocated by the client</li> <li>• An attempt to store in a read-only or unallocated color map entry</li> <li>• An attempt to modify the access control list from other than the local host</li> <li>• An attempt to select an event type, which only one client can select at a time, that has already been selected</li> </ul> |
| XSC_BAD_ALLOC          | The server did not allocate the requested resource for any cause.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                         |
| XSC_BAD_ATOM           | The value specified in an atom argument does not name a defined atom.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     |
| XSC_BAD_COLOR          | A value specified for a color map argument does not name a defined color map.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                             |
| XSC_BAD_CURSOR         | A value specified for a cursor argument does not name a defined cursor.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
| XSC_BAD_DRAWABLE       | A value specified for a drawable argument does not name a defined window or pixmap.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                       |
| XSC_BAD_FONT           | A value specified for a font argument does not name a defined font (or, in some cases, graphics context).                                                                                                                                                                                                                                                                                                                                                                                                                                                                 |
| XSC_BAD_GC             | A value specified for a graphics context argument does not name a defined graphics context.                                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| XSC_BAD_ID_CHOICE      | The value you choose for a resource identifier that either is not included in the range assigned to the client, or is already in use. Normally cannot occur and should be considered a server or Xlib error.                                                                                                                                                                                                                                                                                                                                                              |
| XSC_BAD_IMPLEMENTATION | The server does not implement the entire request. Most likely caused by a server extension; a server that generates this error for a core protocol request is deficient.                                                                                                                                                                                                                                                                                                                                                                                                  |
| XSC_BAD_LENGTH         | Request is shorter or longer than required to minimally contain the arguments. Usually indicates an internal Xlib or server error. The length of a request exceeds the maximum length accepted by the server.                                                                                                                                                                                                                                                                                                                                                             |
| XSC_BAD_MATCH          | <p>Possible causes are as follows:</p> <ul style="list-style-type: none"> <li>• In a graphics request, the root and depth of the graphics context do not match those of the drawable.</li> <li>• An input-only window is used as a drawable.</li> <li>• One argument or pair of arguments has the correct type and range but fails to match in some other way required by the request.</li> <li>• An input-only window lacks this attribute.</li> </ul>                                                                                                                   |
| XSC_BAD_NAME           | The font or color that you specified does not exist.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      |

(continued on next page)



**Table 3–68 (Cont.) Xlib Error Codes**

| <b>Error Code</b> | <b>Description</b>                                                                                                                                                                                                                                           |
|-------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| XSC_BAD_PIXMAP    | A value that you specified for a pixmap argument does not name a defined pixmap.                                                                                                                                                                             |
| XSC_BAD_REQUEST   | The major or minor op-code that you specified does not indicate a valid request. This is usually an Xlib or server error.                                                                                                                                    |
| XSC_BAD_VALUE     | Some numeric values fall outside the range of values accepted by the request. Unless a range is specified for an argument, the full range defined by the argument's type is accepted. Any argument defined as a set of alternatives can generate this error. |
| XSC_BAD_WINDOW    | Value specified for a window argument does not name a defined window.                                                                                                                                                                                        |



# Part II

---

## OpenVMS DECwindows Intrinsic

Part II documents intrinsic routines and data structures and includes the following chapters:

- Chapter 4 – Intrinsic Routines
- Chapter 5 – Intrinsic Data Structures



---

## Intrinsics Routines

This chapter documents the intrinsics routines. See Section 1.2 for information about the format used to describe each routine.

---

### XtAddActions

#### FORMAT

```
XtAddActions
    (action, num_actions)
```

#### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| action      | XtActionList | read   | value     |
| num_actions | Cardinal     | read   | value     |

This routine, superseded by `XtAppAddActions`, is supported for backward compatibility.

---

### XtAddCallback

#### FORMAT

```
XtAddCallback
    (w, callback_name, callback, client_data)
```

#### Argument Information

| Argument      | Type           | Access | Mechanism |
|---------------|----------------|--------|-----------|
| w             | Widget         | read   | value     |
| callback_name | XtString       | read   | reference |
| callback      | XtCallbackProc | read   | value     |
| client_data   | XtPointer      | read   | value     |

## Intrinsics Routines

### XtAddCallbacks

---

## XtAddCallbacks

### FORMAT

XtAddCallbacks  
(w, callback\_name, callbacks)

### Argument Information

| Argument      | Type                  | Access | Mechanism |
|---------------|-----------------------|--------|-----------|
| w             | Widget                | read   | value     |
| callback_name | XtString              | read   | reference |
| callbacks     | XtCallbackRec (array) | read   | reference |

---

## XtAddConverter

### FORMAT

XtAddConverter  
(from\_type, to\_type, converter, convert\_args, num\_args)

### Argument Information

| Argument     | Type                    | Access | Mechanism |
|--------------|-------------------------|--------|-----------|
| from_type    | XtString                | read   | reference |
| to_type      | XtString                | read   | reference |
| converter    | XtConverter             | read   | value     |
| convert_args | XtConvertArgRec (array) | read   | reference |
| num_args     | Cardinal                | read   | value     |

---

This routine, superseded by XtSetTypeConverter and XtAppSetTypeConverter, is supported for backward compatibility.

---

## XtAddEventHandler

### FORMAT

XtAddEventHandler  
(w, event\_mask, nonmaskable, proc, client\_data)

**Argument Information**

| <b>Argument</b> | <b>Type</b>    | <b>Access</b> | <b>Mechanism</b> |
|-----------------|----------------|---------------|------------------|
| w               | Widget         | read          | value            |
| event_mask      | EventMask      | read          | value            |
| nonmaskable     | Boolean        | read          | value            |
| proc            | XtEventHandler | read          | value            |
| client_data     | XtPointer      | read          | value            |

**XtAddExposureToRegion**

**FORMAT**

```
XtAddExposureToRegion
    (event, region)
```

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| event           | Event       | read          | reference        |
| region          | Region      | read          | value            |

**XtAddGrab**

**FORMAT**

```
XtAddGrab
    (w, exclusive, spring_loaded)
```

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| w               | Widget      | read          | value            |
| exclusive       | Boolean     | read          | value            |
| spring_loaded   | Boolean     | read          | value            |

## Intrinsics Routines

### XtAddInput

---

## XtAddInput

### FORMAT

result XtAddInput  
(source, condition, proc, client\_data)

### Argument Information

| Argument    | Type                | Access | Mechanism |
|-------------|---------------------|--------|-----------|
| result      | XtInputId           | write  | value     |
| source      | LONGWORD            | read   | value     |
| condition   | XtPointer           | read   | value     |
| proc        | XtInputCallbackProc | read   | value     |
| client_data | XtPointer           | read   | value     |

This routine, superseded by XtAppAddInput, is supported for backward compatibility.

---

## XtAddRawEventHandler

### FORMAT

XtAddRawEventHandler  
(w, event\_mask, nonmaskable, proc, client\_data)

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| w           | Widget         | read   | value     |
| event_mask  | EventMask      | read   | value     |
| nonmaskable | Boolean        | read   | value     |
| proc        | XtEventHandler | read   | value     |
| client_data | XtPointer      | read   | value     |



## XtAddTimeOut

### FORMAT

result XtAddTimeOut  
(interval, proc, client\_data)

### Argument Information

| Argument    | Type                | Access | Mechanism |
|-------------|---------------------|--------|-----------|
| result      | XtIntervalId        | write  | value     |
| interval    | LONGWORD UNSIGNED   | read   | value     |
| proc        | XtTimerCallbackProc | read   | value     |
| client_data | XtPointer           | read   | value     |

This routine, superseded by XtAppAddTimeOut, is supported for backward compatibility.

## XtAddWorkProc

### FORMAT

result XtAddWorkProc  
(proc, client\_data)

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| result      | XtWorkProcId | write  | value     |
| proc        | XtPointer    | read   | value     |
| client_data | XtPointer    | read   | value     |

This routine, superseded by XtAppAddWorkProc, is supported for backward compatibility.

## XtAllocateGC

### FORMAT

result XtAllocateGC  
(widget, depth, valueMask, values, dynamicMask, unusedMask)

## Intrinsics Routines

### XtAllocateGC

#### Argument Information

| Argument    | Type     | Access | Mechanism |
|-------------|----------|--------|-----------|
| result      | GC       | write  | value     |
| widget      | Widget   | read   | value     |
| depth       | Cardinal | read   | value     |
| valueMask   | XtGCMask | read   | value     |
| values      | XGCValue | read   | reference |
| dynamicMask | XtGCMask | read   | value     |
| unusedMask  | XtGCMask | read   | value     |

---

## XtAppAddActionHook

### FORMAT

result XtAppAddActionHook  
(app, proc, client\_data)

#### Argument Information

| Argument    | Type             | Access | Mechanism |
|-------------|------------------|--------|-----------|
| result      | XtActionHookId   | write  | value     |
| app         | XtAppContext     | read   | value     |
| proc        | XtActionHookProc | read   | value     |
| client_data | XtPointer        | read   | value     |

---

## XtAppAddActions

### FORMAT

XtAppAddActions  
(app\_context, actions, num\_actions)

#### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | value     |
| actions     | XtActionList | read   | value     |
| num_actions | Cardinal     | read   | value     |

## XtAppAddConverter

### FORMAT

XtAppAddConverter  
(app\_context, from\_type, toType, converter, convert\_args, num\_args)

### Argument Information

| Argument     | Type                    | Access | Mechanism |
|--------------|-------------------------|--------|-----------|
| app_context  | XtAppContext            | read   | reference |
| from_type    | XtString                | read   | reference |
| toType       | XtString                | read   | reference |
| converter    | XtConverter             | read   | value     |
| convert_args | XtConvertArgRec (array) | read   | reference |
| num_args     | Cardinal                | read   | value     |

This routine, superseded by XtSetTypeConverter and XtAppSetTypeConverter, is supported for backward compatibility.

## XtAppAddInput

### FORMAT

result XtAppAddInput  
(app\_context, source, condition, proc, client\_data)

### Argument Information

| Argument    | Type                | Access | Mechanism |
|-------------|---------------------|--------|-----------|
| result      | XtInputId           | write  | value     |
| app_context | XtAppContext        | read   | value     |
| source      | LONGWORD            | read   | value     |
| condition   | XtPointer           | read   | value     |
| proc        | XtInputCallbackProc | read   | value     |
| client_data | XtPointer           | read   | value     |

## Intrinsics Routines

### XtAppAddTimeOut

---

## XtAppAddTimeOut

### FORMAT

result XtAppAddTimeOut  
(app\_context, interval, proc, client\_data)

### Argument Information

| Argument    | Type                | Access | Mechanism |
|-------------|---------------------|--------|-----------|
| result      | XtIntervalId        | write  | value     |
| app_context | XtAppContext        | read   | value     |
| interval    | LONGWORD UNSIGNED   | read   | value     |
| proc        | XtTimerCallbackProc | read   | value     |
| client_data | XtPointer           | read   | value     |

---

## XtAppAddWorkProc

### FORMAT

result XtAppAddWorkProc  
(app\_context, proc, client\_data)

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| result      | XtWorkProc   | write  | value     |
| app_context | XtAppContext | read   | value     |
| proc        | XtWorkProc   | read   | value     |
| client_data | XtPointer    | read   | value     |

---

## XtAppCreateShell

### FORMAT

widget XtAppCreateShell  
(application\_name, application\_class, widget\_class, display, args, num\_args)

Argument Information

| Argument          | Type        | Access | Mechanism |
|-------------------|-------------|--------|-----------|
| widget            | Widget      | write  | value     |
| application_name  | XtString    | read   | reference |
| application_class | XtString    | read   | reference |
| widget_class      | WidgetClass | read   | value     |
| display           | Display     | read   | value     |
| args              | Arg (array) | read   | reference |
| num_args          | Cardinal    | read   | value     |

---

## XtAppError

### FORMAT

XtAppError  
(app\_context, message)

Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | value     |
| message     | XtString     | read   | reference |

---

## XtAppErrorMsg

### FORMAT

XtAppErrorMsg  
(app\_context, name, type, class, default, params, num\_params)

Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | reference |
| name        | XtString     | read   | reference |
| type        | XtString     | read   | reference |
| class       | XtString     | read   | reference |
| default     | XtString     | read   | reference |

## Intrinsics Routines

### XtAppErrorMsg

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| params     | XtString (array) | read   | reference |
| num_params | Cardinal         | read   | reference |

---

## XtAppGetErrorDatabase

### FORMAT

result XtAppGetErrorDatabase  
(app\_context)

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| result      | XrmDatabase  | write  | value     |
| app_context | XtAppContext | read   | value     |

---

## XtAppGetErrorDatabaseText

### FORMAT

XtAppGetErrorDatabaseText  
(app\_context, name, type, class, default, buffer\_return, nbytes, database)

### Argument Information

| Argument      | Type         | Access | Mechanism |
|---------------|--------------|--------|-----------|
| app_context   | XtAppContext | read   | value     |
| name          | XtString     | read   | reference |
| type          | XtString     | read   | reference |
| class         | XtString     | read   | reference |
| default       | XtString     | read   | reference |
| buffer_return | XtString     | read   | reference |
| nbytes        | LONGWORD     | read   | value     |
| database      | XrmDatabase  | read   | value     |

## XtAppGetSelectionTimeout

### FORMAT

result XtAppGetSelectionTimeout  
(app\_context)

### Argument Information

| Argument    | Type              | Access | Mechanism |
|-------------|-------------------|--------|-----------|
| result      | LONGWORD UNSIGNED | write  | value     |
| app_context | XtAppContext      | read   | value     |

## XtAppInitialize

### FORMAT

XtAppInitialize  
(app\_context\_return, application\_class, options, num\_options, argc\_in\_out,  
argv\_in\_out, fallback\_resources, args, num\_args)

### Argument Information

| Argument           | Type                         | Access | Mechanism |
|--------------------|------------------------------|--------|-----------|
| app_context_return | XtAppContext                 | write  | value     |
| application_class  | XtString                     | read   | reference |
| options            | LONGWORD UNSIGNED<br>(array) | read   | reference |
| num_options        | Cardinal                     | read   | value     |
| argc_in_out        | Cardinal                     | write  | reference |
| argv_in_out        | XtString (array)             | write  | reference |
| fallback_resources | XtString (array)             | read   | reference |
| args               | Arg (array)                  | read   | reference |
| num_args           | Cardinal                     | read   | value     |

## Intrinsics Routines

### XtAppMainLoop

---

## XtAppMainLoop

### FORMAT

XtAppMainLoop  
(app\_context )

### Argument Information

---

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | value     |

---

## XtAppNextEvent

### FORMAT

XtAppNextEvent  
(app\_context, event\_return)

### Argument Information

---

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| app_context  | XtAppContext | read   | value     |
| event_return | Event        | write  | reference |

---

## XtAppPeekEvent

### FORMAT

result XtAppPeekEvent  
(app\_context, event\_return)

### Argument Information

---

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Boolean      | write  | value     |
| app_context  | XtAppContext | read   | value     |
| event_return | Event        | read   | reference |

---



## XtAppPending

### FORMAT

result XtAppPending  
(app\_context)

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| result      | XtInputMask  | write  | value     |
| app_context | XtAppContext | read   | value     |

## XtAppProcessEvent

### FORMAT

XtAppProcessEvent  
(app\_context, mask)

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | value     |
| mask        | XtInputMask  | read   | value     |

## XtAppReleaseCacheRefs

### FORMAT

XtAppReleaseCacheRefs  
(app\_context, cache\_ref)

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | value     |
| cache_ref   | XtCacheRef   | read   | reference |

## Intrinsics Routines

### XtAppSetErrorHandler

---

## XtAppSetErrorHandler

### FORMAT

XtAppSetErrorHandler  
(app\_context, handler)

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| app_context | XtAppContext   | read   | value     |
| handler     | XtErrorHandler | read   | value     |

---

## XtAppSetErrorMsgHandler

### FORMAT

result XtAppSetErrorMsgHandler  
(app\_context, msg\_handler)

### Argument Information

| Argument    | Type              | Access | Mechanism |
|-------------|-------------------|--------|-----------|
| result      | XtErrorMsgHandler | write  | value     |
| app_context | XtAppContext      | read   | value     |
| msg_handler | XtErrorMsgHandler | read   | value     |

---

## XtAppSetFallbackResources

### FORMAT

XtAppSetFallbackResources  
(app\_context, specification\_list)

### Argument Information

| Argument           | Type             | Access | Mechanism |
|--------------------|------------------|--------|-----------|
| app_context        | XtAppContext     | read   | value     |
| specification_list | XtString (array) | read   | reference |

---

## XtAppSetSelectionTimeout

### FORMAT

XtAppSetSelectionTimeout  
(app\_context, timeout)

### Argument Information

| Argument    | Type              | Access | Mechanism |
|-------------|-------------------|--------|-----------|
| app_context | XtAppContext      | read   | value     |
| timeout     | LONGWORD UNSIGNED | read   | value     |

---

## XtAppSetTypeConverter

### FORMAT

XtAppSetTypeConverter  
(app\_context, from\_type, toType, converter, convert\_args, num\_args, cache\_type, destructor)

### Argument Information

| Argument     | Type                    | Access | Mechanism |
|--------------|-------------------------|--------|-----------|
| app_context  | XtAppContext            | read   | value     |
| from_type    | XtString                | read   | reference |
| toType       | XtString                | read   | reference |
| converter    | XtTypeConverter         | read   | value     |
| convert_args | XtConvertArgRec (array) | read   | reference |
| num_args     | Cardinal                | read   | value     |
| cache_type   | XtCacheType             | read   | value     |
| destructor   | XtDestructor            | read   | value     |

## Intrinsics Routines

### XtAppSetWarningHandler

---

## XtAppSetWarningHandler

### FORMAT

XtAppSetWarningHandler  
(app\_context, handler)

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| app_context | XtAppContext   | read   | value     |
| handler     | XtErrorHandler | read   | value     |

---

## XtAppSetWarningMsgHandler

### FORMAT

XtAppSetWarningMsgHandler  
(app\_context, msg\_handler)

### Argument Information

| Argument    | Type              | Access | Mechanism |
|-------------|-------------------|--------|-----------|
| app_context | XtAppContext      | read   | value     |
| msg_handler | XtErrorMsgHandler | read   | value     |

---

## XtAppWarning

### FORMAT

XtAppWarning  
(app\_context, message )

### Argument Information

| Argument    | Type         | Access | Mechanism |
|-------------|--------------|--------|-----------|
| app_context | XtAppContext | read   | value     |
| message     | XtString     | read   | reference |

---

## XtAppWarningMsg

### FORMAT

XtAppWarningMsg  
(app\_context, name, type, class, default, params, num\_params)

### Argument Information

| Argument    | Type             | Access | Mechanism |
|-------------|------------------|--------|-----------|
| app_context | XtAppContext     | read   | value     |
| name        | XtString         | read   | reference |
| type        | XtString         | read   | reference |
| class       | XtString         | read   | reference |
| default     | XtString         | read   | reference |
| params      | XtString (array) | read   | reference |
| num_params  | Cardinal         | read   | reference |

---

## XtAugmentTranslations

### FORMAT

XtAugmentTranslations  
(w, translations)

### Argument Information

| Argument     | Type           | Access | Mechanism |
|--------------|----------------|--------|-----------|
| w            | Widget         | read   | value     |
| translations | XtTranslations | read   | value     |

---

## XtBuildEventMask

### FORMAT

result XtBuildEventMask  
(w)

## Intrinsics Routines

### XtBuildEventMask

#### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| result   | EventMask | write  | value     |
| w        | Widget    | read   | value     |

---

## XtCallAcceptFocus

### FORMAT

result XtCallAcceptFocus  
(w, time)

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| w        | Widget  | read   | value     |
| time     | Time    | read   | reference |

---

## XtCallActionProc

### FORMAT

XtCallActionProc  
(widget, action, event, params, num\_params)

#### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| widget     | Widget           | read   | value     |
| action     | XtString         | read   | value     |
| event      | Event            | read   | reference |
| params     | XtString (array) | read   | reference |
| num_params | Cardinal         | read   | value     |

---

## XtCallbackExclusive

### FORMAT

XtCallbackExclusive  
(w, client\_data, callback\_data)

### Argument Information

| Argument      | Type      | Access | Mechanism |
|---------------|-----------|--------|-----------|
| w             | Widget    | read   | value     |
| client_data   | XtPointer | read   | value     |
| callback_data | XtPointer | read   | value     |

---

## XtCallbackNone

### FORMAT

XtCallbackNone  
(w, client\_data, call\_data)

### Argument Information

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| w           | Widget    | read   | value     |
| client_data | XtPointer | read   | value     |
| call_data   | XtPointer | read   | value     |

---

## XtCallbackNonexclusive

### FORMAT

XtCallbackNonexclusive  
(w, client\_data, call\_data)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

## Intrinsics Routines

### XtCallbackNonexclusive

---

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| client_data | XtPointer | read   | value     |
| call_data   | XtPointer | read   | value     |

---

### XtCallbackPopdown

#### FORMAT

XtCallbackPopdown  
(w, client\_data, call\_data)

#### Argument Information

---

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| w           | Widget    | read   | value     |
| client_data | XtPointer | read   | value     |
| call_data   | XtPointer | read   | value     |

---

### XtCallbackReleaseCacheRef

#### FORMAT

XtCallbackReleaseCacheRef  
(object, client\_data, call\_data)

#### Argument Information

---

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| object      | Widget    | read   | value     |
| client_data | XtPointer | read   | value     |
| call_data   | XtPointer | read   | value     |

---



---

## XtCallbackReleaseCacheRefList

### FORMAT

XtCallbackReleaseCacheRefList  
(object, client\_data, call\_data)

### Argument Information

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| object      | Widget    | read   | value     |
| client_data | XtPointer | read   | value     |
| call_data   | XtPointer | read   | value     |

---

## XtCallCallbackList

### FORMAT

XtCallCallbackList  
(widget, callbacks, call\_data)

### Argument Information

| Argument  | Type                  | Access | Mechanism |
|-----------|-----------------------|--------|-----------|
| widget    | Widget                | read   | value     |
| callbacks | XtCallbackRec (array) | read   | reference |
| call_data | XtPointer             | read   | value     |

---

## XtCallCallbacks

### FORMAT

XtCallCallbacks  
(w, callback\_name, call\_data)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

## Intrinsics Routines

### XtCallCallbacks

---

| Argument      | Type      | Access | Mechanism |
|---------------|-----------|--------|-----------|
| callback_name | XtString  | read   | reference |
| call_data     | XtPointer | read   | value     |

---

## XtCallConverter

### FORMAT

result XtCallConverter

(display, converter, args, num\_args, from, to\_in\_out, cache\_ref\_return)

### Argument Information

---

| Argument         | Type            | Access | Mechanism |
|------------------|-----------------|--------|-----------|
| result           | Boolean         | write  | value     |
| display          | Display         | read   | value     |
| converter        | XtTypeConverter | read   | value     |
| args             | XrmValuePtr     | read   | value     |
| num_args         | Cardinal        | read   | value     |
| from             | XrmValuePtr     | read   | value     |
| to_in_out        | XrmValuePtr     | write  | reference |
| cache_ref_return | XtCacheRef      | write  | reference |

---

## XtCalloc

### FORMAT

result XtCalloc

(num, size)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | ADDRESS  | write  | value     |
| num      | Cardinal | read   | value     |
| size     | Cardinal | read   | value     |

---

## XtClass

### FORMAT

result XtClass  
(w)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | WidgetClass | write  | value     |
| w        | Widget      | read   | value     |

## XtCloseDisplay

### FORMAT

XtCloseDisplay  
(display)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| display  | Display | read   | value     |

## XtConvert

### FORMAT

XtConvert  
(w, from\_type, from, to\_type, to\_return)

### Argument Information

| Argument  | Type        | Access | Mechanism |
|-----------|-------------|--------|-----------|
| w         | Widget      | read   | value     |
| from_type | XtString    | read   | reference |
| from      | XrmValuePtr | read   | value     |
| to_type   | XtString    | read   | reference |

## Intrinsics Routines

### XtConvert

| Argument  | Type        | Access | Mechanism |
|-----------|-------------|--------|-----------|
| to_return | XrmValuePtr | read   | value     |

This routine, superseded by `XtConvertAndStore`, is supported for backward compatibility.

---

## XtConvertAndStore

### FORMAT

status XtConvertAndStore  
(object, from\_type, from, toType, to\_in\_out)

### Argument Information

| Argument  | Type        | Access | Mechanism |
|-----------|-------------|--------|-----------|
| status    | Boolean     | write  | value     |
| object    | Widget      | read   | value     |
| from_type | XtString    | read   | reference |
| from      | XrmValuePtr | read   | value     |
| toType    | XtString    | read   | reference |
| to_in_out | XrmValuePtr | read   | value     |

---

## XtConvertCase

### FORMAT

XtConvertCase  
(display, keysym, lower\_return, upper\_return)

### Argument Information

| Argument     | Type    | Access | Mechanism |
|--------------|---------|--------|-----------|
| display      | Display | read   | value     |
| keysym       | KeySym  | read   | value     |
| lower_return | KeySym  | write  | reference |
| upper_return | KeySym  | write  | reference |

---

## XtCreateApplicationContext

### FORMAT

result XtCreateApplicationContext  
(result)

### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| result   | XtAppContext | write  | value     |

---

## XtCreateApplicationShell

### FORMAT

widget XtCreateApplicationShell  
(name, widget\_class, args, num\_args)

### Argument Information

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| widget       | Widget      | write  | value     |
| name         | XtString    | read   | reference |
| widget_class | WidgetClass | read   | value     |
| args         | Arg (array) | read   | reference |
| num_args     | Cardinal    | read   | value     |

This routine, superseded by XtAppCreateShell, is supported for backward compatibility.

---

## XtCreateManagedWidget

### FORMAT

widget XtCreateManagedWidget  
(name, widget\_class, parent, args, num\_args)

## Intrinsics Routines

### XtCreateManagedWidget

#### Argument Information

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| widget       | Widget      | write  | value     |
| name         | XtString    | read   | reference |
| widget_class | WidgetClass | read   | value     |
| parent       | Widget      | read   | value     |
| args         | Arg (array) | read   | reference |
| num_args     | Cardinal    | read   | value     |

---

## XtCreatePopupShell

### FORMAT

widget XtCreatePopupShell  
(name, widget\_class, parent, args, num\_args)

#### Argument Information

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| widget       | Widget      | write  | value     |
| name         | XtString    | read   | reference |
| widget_class | WidgetClass | read   | value     |
| parent       | Widget      | read   | value     |
| args         | Arg (array) | write  | value     |
| num_args     | Cardinal    | read   | value     |

---

## XtCreateWidget

### FORMAT

widget XtCreateWidget  
(name, object\_class, parent, args, num\_args)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | return    |
| name     | XtString | read   | value     |

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| object_class    | WidgetClass | read          | value            |
| parent          | Widget      | read          | value            |
| args            | Arg (array) | read          | reference        |
| num_args        | Cardinal    | read          | value            |

---

## **XtCvtColorToPixel**

### **FORMAT**

result XtCvtColorToPixel  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### **Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | Boolean     | write         | value            |
| display         | Display     | read          | value            |
| args            | XrmValuePtr | read          | reference        |
| num_args        | Cardinal    | read          | value            |
| fromVal         | XrmValuePtr | read          | value            |
| toVal           | XrmValuePtr | read          | value            |
| closure_ret     | XtPointer   | write         | reference        |

---

## **XtCvtIntToBool**

### **FORMAT**

result XtCvtIntToBool  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### **Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | Boolean     | write         | value            |
| display         | Display     | read          | value            |
| args            | XrmValuePtr | read          | reference        |
| num_args        | Cardinal    | read          | value            |
| fromVal         | XrmValuePtr | read          | value            |

## Intrinsics Routines

### XtCvtIntToBool

---

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtIntToBoolean

### FORMAT

result XtCvtIntToBoolean  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

---

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtIntToColor

### FORMAT

result XtCvtIntToColor  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

---

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---



## XtCvtIntToFloat

### FORMAT

```
result XtCvtIntToFloat
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## XtCvtIntToFont

### FORMAT

```
result XtCvtIntToFont
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## Intrinsics Routines

### XtCvtIntToPixel

---

## XtCvtIntToPixel

### FORMAT

result XtCvtIntToPixel  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtIntToPixmap

### FORMAT

result XtCvtIntToPixmap  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## XtCvtIntToShort

### FORMAT

```
result XtCvtIntToShort
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## XtCvtIntToUnsignedChar

### FORMAT

```
result XtCvtIntToUnsignedChar
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## Intrinsics Routines

### XtCvtStringToAcceleratorTable

---

## XtCvtStringToAcceleratorTable

### FORMAT

result XtCvtStringToAcceleratorTable  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToAtom

### FORMAT

result XtCvtStringToAtom  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

---

## XtCvtStringToBool

### FORMAT

result XtCvtStringToBool  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToBoolean

### FORMAT

result XtCvtStringToBoolean  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## Intrinsics Routines

### XtCvtStringToCursor

---

## XtCvtStringToCursor

### FORMAT

result XtCvtStringToCursor  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToDimension

### FORMAT

result XtCvtStringToDimension  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToDisplay

### FORMAT

```
result XtCvtStringToDisplay
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToFile

### FORMAT

```
result XtCvtStringToFile
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## Intrinsics Routines

### XtCvtStringToFloat

---

## XtCvtStringToFloat

### FORMAT

result XtCvtStringToFloat  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToFont

### FORMAT

result XtCvtStringToFont  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |



---

## XtCvtStringToFontSet

### FORMAT

```
result XtCvtStringToFontSet
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToFontStruct

### FORMAT

```
result XtCvtStringToFontStruct
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## Intrinsics Routines

### XtCvtStringToInt

---

## XtCvtStringToInt

### FORMAT

result XtCvtStringToInt  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToInitialState

### FORMAT

result XtCvtStringToInitialState  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToPixel

### FORMAT

```
result XtCvtStringToPixel
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToShort

### FORMAT

```
result XtCvtStringToShort
    (display, args, num_args, fromVal, toVal, closure_ret)
```

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

## Intrinsics Routines

### XtCvtStringToTranslationTable

---

## XtCvtStringToTranslationTable

### FORMAT

result XtCvtStringToTranslationTable  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

## XtCvtStringToUnsignedChar

### FORMAT

result XtCvtStringToUnsignedChar  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| result      | Boolean     | write  | value     |
| display     | Display     | read   | value     |
| args        | XrmValuePtr | read   | reference |
| num_args    | Cardinal    | read   | value     |
| fromVal     | XrmValuePtr | read   | value     |
| toVal       | XrmValuePtr | read   | value     |
| closure_ret | XtPointer   | write  | reference |

---

---

## **XtCvtStringToVisual**

### **FORMAT**

result XtCvtStringToVisual  
(display, args, num\_args, fromVal, toVal, closure\_ret)

### **Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | Boolean     | write         | value            |
| display         | Display     | read          | value            |
| args            | XrmValuePtr | read          | reference        |
| num_args        | Cardinal    | read          | value            |
| fromVal         | XrmValuePtr | read          | value            |
| toVal           | XrmValuePtr | read          | value            |
| closure_ret     | XtPointer   | write         | reference        |

---

## **XtDatabase**

### **FORMAT**

result XtDatabase  
(display)

### **Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | XrmDatabase | write         | value            |
| display         | Display     | read          | value            |

---

## **XtDestroyApplicationContext**

### **FORMAT**

XtDestroyApplicationContext  
(app\_context)

## Intrinsics Routines

### XtDestroyApplicationContext

#### Argument Information

| Argument    | Type                 | Access | Mechanism |
|-------------|----------------------|--------|-----------|
| app_context | XtApplicationContext | read   | value     |

---

## XtDestroyGC

### FORMAT

XtDestroyGC  
(w, gc )

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |
| gc       | GC     | read   | value     |

This routine, superseded by XtReleaseGC, is supported for backward compatibility.

---

## XtDestroyWidget

### FORMAT

XtDestroyWidget  
(w)

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

## XtDirectConvert

### FORMAT

XtDirectConvert  
(converter, args, num\_args, from, to\_return)

### Argument Information

| Argument  | Type        | Access | Mechanism |
|-----------|-------------|--------|-----------|
| converter | XtConverter | read   | value     |
| args      | XrmValuePtr | read   | reference |
| num_args  | Cardinal    | read   | value     |
| from      | XrmValuePtr | read   | value     |
| to_return | XrmValuePtr | read   | value     |

This routine, superseded by XtCallConverter, is supported for backward compatibility.

## XtDisownSelection

### FORMAT

XtDisownSelection  
(w, selection, time)

### Argument Information

| Argument  | Type   | Access | Mechanism |
|-----------|--------|--------|-----------|
| w         | Widget | read   | value     |
| selection | Atom   | read   | value     |
| time      | Time   | read   | value     |

## XtDispatchEvent

### FORMAT

result XtDispatchEvent  
(event)

## Intrinsics Routines

### XtDispatchEvent

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| event    | Event   | read   | reference |

---

## XtDisplay

### FORMAT

result XtDisplay  
(w)

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Display | write  | value     |
| w        | Widget  | read   | value     |

---

## XtDisplayInitialize

### FORMAT

XtDisplayInitialize  
(app\_context, display, application\_name, application\_class, options, num\_options,  
args, num\_args)

#### Argument Information

| Argument          | Type              | Access | Mechanism |
|-------------------|-------------------|--------|-----------|
| app_context       | XtAppContext      | read   | value     |
| display           | Display           | read   | value     |
| application_name  | XtString          | read   | reference |
| application_class | XtString (array)  | read   | reference |
| options           | LONGWORD UNSIGNED | read   | reference |
| num_options       | Cardinal          | read   | value     |
| args              | Cardinal          | modify | reference |
| num_args          | ADDRESS (array)   | modify | reference |



---

## XtDisplayOfObject

### FORMAT

result XtDisplayOfObject  
(object)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Display | write  | value     |
| object   | Widget  | read   | value     |

---

## XtDisplayStringConvWarning

### FORMAT

XtDisplayStringConvWarning  
(display, from\_value, toType)

### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| display    | Display  | read   | value     |
| from_value | XtString | read   | reference |
| toType     | XtString | read   | reference |

---

## XtDisplayToApplicationContext

### FORMAT

return XtDisplayToApplicationContext  
(display)

### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| return   | XtAppContext | write  | value     |
| display  | Display      | read   | value     |

## Intrinsics Routines

### XtError

---

## XtError

### FORMAT

XtError  
(message)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| message  | XtString | read   | reference |

This routine, superseded by `XtAppErrorMsg`, is supported for backward compatibility.

---

## XtErrorMsg

### FORMAT

XtErrorMsg  
(name, type, class, default, params, num\_params)

### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| name       | XtString         | read   | reference |
| type       | XtString         | read   | reference |
| class      | XtString         | read   | reference |
| default    | XtString         | read   | reference |
| params     | XtString (array) | read   | reference |
| num_params | Cardinal         | read   | reference |

This routine, superseded by `XtAppErrorMsg`, is supported for backward compatibility.

---

## XtFindFile

### FORMAT

result XtFindFile  
(path, substitutions, num\_substitutions, predicate)

**Argument Information**

| <b>Argument</b>   | <b>Type</b>     | <b>Access</b> | <b>Mechanism</b> |
|-------------------|-----------------|---------------|------------------|
| result            | XtString        | write         | value            |
| path              | XtString        | read          | reference        |
| substitutions     | Substitution    | read          | reference        |
| num_substitutions | Cardinal        | read          | value            |
| predicate         | XtFilePredicate | read          | value            |

**XtFree**

**FORMAT**

XtFree  
(ptr)

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| ptr             | ADDRESS     | read          | value            |

**XtGetActionList**

**FORMAT**

XtGetActionList  
(widget\_class,actions\_return,num\_actions\_return)

**Argument Information**

| <b>Argument</b>    | <b>Type</b>  | <b>Access</b> | <b>Mechanism</b> |
|--------------------|--------------|---------------|------------------|
| widget_class       | WidgetClass  | read          | value            |
| actions_return     | XtActionList | write         | reference        |
| num_actions_return | Cardinal     | write         | reference        |

## Intrinsics Routines

### XtGetActionKeysym

---

## XtGetActionKeysym

### FORMAT

result XtGetActionKeysym  
(event, modifiers\_return)

### Argument Information

| Argument         | Type      | Access | Mechanism |
|------------------|-----------|--------|-----------|
| result           | KeySym    | write  | value     |
| event            | Event     | write  | reference |
| modifiers_return | Modifiers | write  | reference |

---

## XtGetApplicationNameAndClass

### FORMAT

XtGetApplicationNameAndClass  
(display, name\_return, class\_return)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| display      | Display  | read   | value     |
| name_return  | XtString | write  | reference |
| class_return | XtString | write  | reference |

---

## XtGetApplicationResources

### FORMAT

XtGetApplicationResources  
(w, base, resources, num\_resources, args, num\_args)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

## Intrinsics Routines

### XtGetApplicationResources

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| base          | XtPointer          | read   | value     |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| args          | Arg (array)        | write  | reference |
| num_args      | Cardinal           | read   | value     |

---

## XtGetConstraintResourceList

### FORMAT

XtGetConstraintResourceList  
(class, resources\_return, num\_resources\_return)

### Argument Information

| Argument             | Type               | Access | Mechanism |
|----------------------|--------------------|--------|-----------|
| class                | WidgetClass        | read   | value     |
| resources_return     | XtResource (array) | write  | reference |
| num_resources_return | Cardinal           | write  | reference |

---

## XtGetErrorDatabase

### FORMAT

result XtGetErrorDatabase

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | XrmDatabase | write  | value     |

This routine, superseded by XtAppGetErrorDatabase, is supported for backward compatibility.

## Intrinsics Routines

### XtGetErrorDatabaseText

---

## XtGetErrorDatabaseText

### FORMAT

XtGetErrorDatabaseText

(name, type, class, default, buffer\_return, nbytes)

### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| name          | XtString | read   | reference |
| type          | XtString | read   | reference |
| class         | XtString | read   | reference |
| default       | XtString | read   | reference |
| buffer_return | XtString | read   | reference |
| nbytes        | LONGWORD | read   | value     |

This routine, superseded by `XtAppGetErrorDatabaseText`, is supported for backward compatibility.

---

## XtGetGC

### FORMAT

result XtGetGC

(object, value\_mask, values)

### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | GC       | write  | value     |
| object     | Widget   | read   | value     |
| value_mask | XtGCMask | read   | value     |
| values     | XGCValue | read   | reference |

---

## XtGetKeysymTable

### FORMAT

result XtGetKeysymTable  
(display, min\_keycode\_return, keysyms\_per\_keycode\_return)

### Argument Information

| Argument                   | Type     | Access | Mechanism |
|----------------------------|----------|--------|-----------|
| result                     | KeySym   | write  | value     |
| display                    | Display  | read   | write     |
| min_keycode_return         | KeyCode  | modify | reference |
| keysyms_per_keycode_return | LONGWORD | write  | reference |

---

## XtGetMultiClickTime

### FORMAT

result XtGetMultiClickTime  
(display)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |

---

## XtGetResourceList

### FORMAT

XtGetResourceList  
(class, resources\_return, num\_resources\_return)

## Intrinsics Routines

### XtGetResourceList

#### Argument Information

| Argument             | Type               | Access | Mechanism |
|----------------------|--------------------|--------|-----------|
| class                | WidgetClass        | read   | value     |
| resources_return     | XtResource (array) | write  | reference |
| num_resources_return | Cardinal           | write  | reference |

---

## XtGetSelectionRequest

### FORMAT

result XtGetSelectionRequest  
(w, selection, request\_id)

#### Argument Information

| Argument   | Type        | Access | Mechanism |
|------------|-------------|--------|-----------|
| result     | ADDRESS     | write  | value     |
| w          | Widget      | read   | value     |
| selection  | Atom        | read   | value     |
| request_id | XtRequestId | read   | value     |

---

## XtGetSelectionTimeout

### FORMAT

result XtGetSelectionTimeout  
( )

#### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| result   | uns longword | write  | value     |

This routine, superseded by XtAppGetSelectionTimeout, is supported for backward compatibility.



---

## XtGetSelectionValue

### FORMAT

XtGetSelectionValue  
(w, selection, target, callback, client\_data, time)

### Argument Information

| Argument    | Type                    | Access | Mechanism |
|-------------|-------------------------|--------|-----------|
| w           | Widget                  | read   | value     |
| selection   | Atom                    | read   | value     |
| target      | Atom                    | read   | value     |
| callback    | XtSelectionCallbackProc | read   | value     |
| client_data | XtPointer               | read   | value     |
| time        | Time                    | read   | value     |

---

## XtGetSelectionValueIncremental

### FORMAT

XtGetSelectionValueIncremental  
(w, selection, target, selection\_callback, client\_data, time)

### Argument Information

| Argument           | Type                    | Access | Mechanism |
|--------------------|-------------------------|--------|-----------|
| w                  | Widget                  | read   | value     |
| selection          | Atom                    | read   | value     |
| target             | Atom                    | read   | value     |
| selection_callback | XtSelectionCallbackProc | read   | value     |
| client_data        | XtPointer               | read   | value     |
| time               | Time                    | read   | value     |

## Intrinsics Routines

### XtGetSelectionValues

---

## XtGetSelectionValues

### FORMAT

XtGetSelectionValues

(w, selection, targets, count, callback, client\_datas, time)

### Argument Information

| Argument     | Type                    | Access | Mechanism |
|--------------|-------------------------|--------|-----------|
| w            | Widget                  | read   | value     |
| selection    | Atom                    | read   | value     |
| targets      | Atom                    | read   | reference |
| count        | longword                | read   | value     |
| callback     | XtSelectionCallbackProc | read   | value     |
| client_datas | XtPointer               | read   | reference |
| time         | Time                    | read   | value     |

---

## XtGetSelectionValuesIncremental

### FORMAT

XtGetSelectionValuesIncremental

(w, selection, targets, count, selection\_callback, client\_data, time)

### Argument Information

| Argument           | Type                    | Access | Mechanism |
|--------------------|-------------------------|--------|-----------|
| w                  | Widget                  | read   | value     |
| selection          | Atom                    | read   | value     |
| targets            | Atom                    | read   | reference |
| count              | longword                | read   | value     |
| selection_callback | XtSelectionCallbackProc | read   | value     |
| client_data        | XtPointer               | read   | reference |
| time               | Time                    | read   | value     |

---

## XtGetSubresources

### FORMAT

XtGetSubresources

(w, base, name, class, resources, num\_resources, args, num\_args)

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| w             | Widget             | read   | value     |
| base          | XtPointer          | read   | value     |
| name          | XtString           | read   | reference |
| class         | XtString           | read   | reference |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| args          | Arg (array)        | read   | reference |
| num_args      | Cardinal           | read   | value     |

---

## XtGetSubvalues

### FORMAT

XtGetSubvalues

(base, resources, num\_resources, args, num\_args)

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| base          | XtPointer          | read   | value     |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| args          | Arg (array)        | read   | reference |
| num_args      | Cardinal           | read   | value     |

## Intrinsics Routines

### XtGetValues

---

## XtGetValues

### FORMAT

XtGetValues  
(object, args, num\_args)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| object   | Widget      | read   | value     |
| args     | Arg (array) | read   | reference |
| num_args | Cardinal    | read   | value     |

---

## XtGrabButton

### FORMAT

XtGrabButton  
(widget, button, modifiers, owner\_events, event\_mask, pointer\_mode,  
keyboard\_mode, confine\_to, cursor)

### Argument Information

| Argument      | Type              | Access | Mechanism |
|---------------|-------------------|--------|-----------|
| widget        | Widget            | read   | value     |
| button        | LONGWORD          | read   | value     |
| modifiers     | Modifiers         | read   | value     |
| owner_events  | Boolean           | read   | value     |
| event_mask    | LONGWORD UNSIGNED | read   | value     |
| pointer_mode  | LONGWORD          | read   | value     |
| keyboard_mode | LONGWORD          | read   | value     |
| confine_to    | Window            | read   | value     |
| cursor        | Cursor            | read   | value     |

---

## XtGrabKey

### FORMAT

XtGrabKey

(widget, keycode, modifiers, owner\_events, pointer\_mode, keyboard\_mode)

### Argument Information

| Argument      | Type      | Access | Mechanism |
|---------------|-----------|--------|-----------|
| widget        | Widget    | read   | value     |
| keycode       | Keycode   | read   | value     |
| modifiers     | Modifiers | read   | value     |
| owner_events  | Boolean   | read   | value     |
| pointer_mode  | LONGWORD  | read   | value     |
| keyboard_mode | LONGWORD  | read   | value     |

## XtGrabKeyboard

### FORMAT

XtGrabKeyboard

(widget, owner\_events, pointer\_mode, keyboard\_mode, time)

### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| widget        | Widget   | read   | value     |
| owner_events  | Boolean  | read   | value     |
| pointer_mode  | LONGWORD | read   | value     |
| keyboard_mode | LONGWORD | read   | value     |
| time          | Time     | read   | value     |

## Intrinsics Routines

### XtGrabPointer

---

## XtGrabPointer

### FORMAT

XtGrabPointer

(widget, owner\_events, event\_mask, pointer\_mode, keyboard\_mode, confine\_to, cursor, time)

### Argument Information

| Argument      | Type              | Access | Mechanism |
|---------------|-------------------|--------|-----------|
| widget        | Widget            | read   | value     |
| owner_events  | Boolean           | read   | value     |
| event_mask    | LONGWORD UNSIGNED | read   | value     |
| pointer_mode  | LONGWORD          | read   | value     |
| keyboard_mode | LONGWORD          | read   | value     |
| confine_to    | Window            | read   | value     |
| cursor        | Cursor            | read   | value     |
| time          | Time              | read   | value     |

---

## XtHasCallbacks

### FORMAT

callback\_status XtHasCallbacks

(w, callback\_name)

### Argument Information

| Argument        | Type             | Access | Mechanism |
|-----------------|------------------|--------|-----------|
| callback_status | XtCallbackStatus | write  | value     |
| w               | Widget           | read   | value     |
| callback_name   | XtString         | read   | reference |

## XtInitialize

### FORMAT

widget XtInitialize  
(shell\_name, application\_class, options, num\_options, argc, argv)

### Argument Information

| Argument          | Type                         | Access | Mechanism |
|-------------------|------------------------------|--------|-----------|
| widget            | Widget                       | write  | value     |
| shell_name        | XtString                     | read   | reference |
| application_class | XtString                     | read   | reference |
| options           | LONGWORD UNSIGNED<br>(array) | read   | value     |
| num_options       | Cardinal                     | read   | value     |
| argc              | Cardinal                     | modify | reference |
| argv              | ADDRESS (array)              | modify | reference |

This routine, superseded by XtAppInitialize, is supported for backward compatibility.

## XtInitializeWidgetClass

### FORMAT

XtInitializeWidgetClass  
(object\_class)

### Argument Information

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| object_class | WidgetClass | read   | value     |

## XtInsertEventHandler

### FORMAT

XtInsertEventHandler  
(w, event\_mask, nonmaskable, proc, client\_data, position)

## Intrinsics Routines

### XtInsertEventHandler

#### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| w           | Widget         | read   | value     |
| event_mask  | EventMask      | read   | value     |
| nonmaskable | Boolean        | read   | value     |
| proc        | XtEventHandler | read   | value     |
| client_data | XtPointer      | read   | value     |
| position    | XtListPosition | read   | value     |

---

## XtInsertRawEventHandler

### FORMAT

XtInsertRawEventHandler

(w, event\_mask, nonmaskable, proc, client\_data, position)

#### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| w           | Widget         | read   | value     |
| event_mask  | EventMask      | read   | value     |
| nonmaskable | Boolean        | read   | value     |
| proc        | XtEventHandler | read   | value     |
| client_data | XtPointer      | read   | value     |
| position    | XtListPosition | read   | value     |

---

## XtInstallAccelerators

### FORMAT

XtInstallAccelerators

(destination, source)

#### Argument Information

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| destination | Widget | read   | value     |
| source      | Widget | read   | value     |



## XtInstallAllAccelerators

### FORMAT

XtInstallAllAccelerators  
(destination, source)

### Argument Information

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| destination | Widget | read   | value     |
| source      | Widget | read   | value     |

## XtIsManaged

### FORMAT

result XtIsManaged  
(w)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| w        | Widget  | read   | value     |

## XtIsObject

### FORMAT

result XtIsObject  
(w)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| w        | Widget  | read   | value     |

## Intrinsics Routines

### XtIsRealized

---

## XtIsRealized

### FORMAT

result XtIsRealized  
(w)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| w        | Widget  | read   | value     |

---

## XtIsSensitive

### FORMAT

result XtIsSensitive  
(w)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| w        | Widget  | read   | value     |

---

## XtIsSubclass

### FORMAT

result XtIsSubclass  
(w, widget\_class )

### Argument Information

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| result       | Boolean     | write  | value     |
| w            | Widget      | read   | value     |
| widget_class | WidgetClass | read   | value     |

---

## XtKeysymToKeycodeList

### FORMAT

XtKeysymToKeycodeList  
(display, keysym, keycodes\_return, keycount\_return)

### Argument Information

| Argument        | Type     | Access | Mechanism |
|-----------------|----------|--------|-----------|
| display         | Display  | read   | value     |
| keysym          | KeySym   | read   | value     |
| keycodes_return | Key      | write  | reference |
| keycount_return | Cardinal | write  | reference |

---

## XtLastTimestampProcessed

### FORMAT

result XtLastTimestampProcessed  
(display)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Time    | write  | value     |
| display  | Display | read   | value     |

---

## XtMainLoop

### FORMAT

XtMainLoop  
( )

This routine, superseded by XtAppMainLoop, is supported for backward compatibility.

## Intrinsics Routines

### XtMakeGeometryRequest

---

## XtMakeGeometryRequest

### FORMAT

status XtMakeGeometryRequest  
(w, requests, reply\_return)

### Argument Information

| Argument     | Type             | Access | Mechanism |
|--------------|------------------|--------|-----------|
| status       | XtGeometryResult | write  | value     |
| w            | Widget           | read   | value     |
| requests     | XtWidgetGeometry | read   | value     |
| reply_return | XtWidgetGeometry | write  | reference |

---

## XtMakeResizeRequest

### FORMAT

status XtMakeResizeRequest  
(w, width, height, width\_return, height\_return)

### Argument Information

| Argument      | Type             | Access | Mechanism |
|---------------|------------------|--------|-----------|
| status        | XtGeometryResult | write  | value     |
| w             | Widget           | read   | value     |
| width         | Dimension        | read   | value     |
| height        | Dimension        | read   | value     |
| width_return  | Dimension        | read   | value     |
| height_return | Dimension        | read   | value     |

---

## XtMalloc

### FORMAT

result XtMalloc  
(size)

Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | ADDRESS  | write  | value     |
| size     | Cardinal | read   | value     |

---

## XtManageChild

### FORMAT

XtManageChild  
(child)

Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| child    | Widget | read   | value     |

---

## XtManageChildren

### FORMAT

XtManageChildren  
(children, num\_children)

Argument Information

| Argument     | Type       | Access | Mechanism |
|--------------|------------|--------|-----------|
| children     | WidgetList | read   | reference |
| num_children | Cardinal   | read   | value     |

---

## XtMenuPopupAction

### FORMAT

XtMenuPopupAction  
(w, event, params, num\_params)

## Intrinsics Routines

### XtMenuPopupAction

#### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| w          | Widget           | read   | value     |
| event      | Event            | read   | reference |
| params     | XtString (array) | read   | reference |
| num_params | Cardinal         | modify | reference |

---

## XtMergeArgLists

### FORMAT

result XtMergeArgLists  
(args, num\_args1, args2, num\_args2)

#### Argument Information

| Argument  | Type        | Access | Mechanism |
|-----------|-------------|--------|-----------|
| result    | ArgList     | write  | value     |
| args      | Arg (array) | read   | reference |
| num_args1 | Cardinal    | read   | value     |
| args2     | Arg (array) | read   | reference |
| num_args2 | Cardinal    | read   | value     |

---

## XtName

### FORMAT

result XtName  
(object)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| object   | Widget   | read   | value     |

---

## XtNameToWidget

### FORMAT

widget XtNameToWidget  
(reference, name)

### Argument Information

| Argument  | Type     | Access | Mechanism |
|-----------|----------|--------|-----------|
| widget    | Widget   | write  | value     |
| reference | Widget   | read   | value     |
| name      | XtString | read   | reference |

---

## XtNextEvent

### FORMAT

XtNextEvent  
(event\_return)

### Argument Information

| Argument     | Type  | Access | Mechanism |
|--------------|-------|--------|-----------|
| event_return | Event | write  | reference |

This routine, superseded by XtAppNextEvent, is supported for backward compatibility.

---

## XtOpenDisplay

### FORMAT

result XtOpenDisplay  
(app\_context, display\_string, application\_name, application\_class, options,  
num\_options, argc, argv)

## Intrinsics Routines

### XtOpenDisplay

#### Argument Information

| Argument          | Type                         | Access | Mechanism |
|-------------------|------------------------------|--------|-----------|
| result            | Display                      | write  | value     |
| app_context       | XtAppContext                 | read   | value     |
| display_string    | XtString                     | read   | reference |
| application_name  | XtString                     | read   | reference |
| application_class | XtString                     | read   | reference |
| options           | LONGWORD UNSIGNED<br>(array) | read   | reference |
| num_options       | Cardinal                     | read   | value     |
| argc              | Cardinal                     | modify | reference |
| argv              | ADDRESS (array)              | modify | reference |

---

## XtOverrideTranslations

### FORMAT

XtOverrideTranslations  
(w, translations)

#### Argument Information

| Argument     | Type           | Access | Mechanism |
|--------------|----------------|--------|-----------|
| w            | Widget         | read   | value     |
| translations | XtTranslations | read   | value     |

---

## XtOwnSelection

### FORMAT

result XtOwnSelection  
(w, selection, time, convert\_proc, lose\_selection, done\_proc)

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| w        | Widget  | read   | value     |



| Argument       | Type                   | Access | Mechanism |
|----------------|------------------------|--------|-----------|
| selection      | Atom                   | read   | value     |
| time           | Time                   | read   | value     |
| convert_proc   | XtConvertSelectionProc | read   | value     |
| lose_selection | XtLoseSelectionProc    | read   | value     |
| done_proc      | XtSelectionDoneProc    | read   | value     |

---

## XtOwnSelectionIncremental

### FORMAT

result XtOwnSelectionIncremental

(w, selection, time, convert\_callback, lose\_callback, done\_callback,  
cancel\_callback, client\_data)

### Argument Information

| Argument         | Type                         | Access | Mechanism |
|------------------|------------------------------|--------|-----------|
| result           | Boolean                      | write  | value     |
| w                | Widget                       | read   | value     |
| selection        | Atom                         | read   | value     |
| time             | Time                         | read   | value     |
| convert_callback | XtConvertSelectionProc       | read   | value     |
| lose_callback    | XtLoseSelectionProc          | read   | value     |
| done_callback    | XtSelectionDoneProc          | read   | value     |
| cancel_callback  | XtCancelConvertSelectionProc | read   | value     |
| client_data      | XtPointer                    | read   | value     |

---

## XtParent

### FORMAT

result XtParent

(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |

## Intrinsics Routines

### XtParent

---

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

---

## XtParseAcceleratorTable

### FORMAT

result XtParseAcceleratorTable  
(source)

### Argument Information

---

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XtAccelerators | write  | value     |
| source   | XtString       | read   | reference |

---

---

## XtParseTranslationTable

### FORMAT

result XtParseTranslationTable  
(source)

### Argument Information

---

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XtTranslations | write  | value     |
| source   | XtString       | read   | reference |

---

---

## XtPeekEvent

### FORMAT

result XtPeekEvent  
(event\_return)

**Argument Information**

| Argument     | Type    | Access | Mechanism |
|--------------|---------|--------|-----------|
| result       | Boolean | write  | value     |
| event_return | Event   | read   | reference |

This routine, superseded by XtAppPeekEvent, is supported for backward compatibility.

**XtPending**

**FORMAT**

```
result XtPending
    ( )
```

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | XtInputMask | write  | value     |

This routine, superseded by XtPending, is supported for backward compatibility.

**XtPopdown**

**FORMAT**

```
XtPopdown
    (popup_shell)
```

**Argument Information**

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| popup_shell | Widget | read   | value     |

## Intrinsics Routines

### XtPopup

---

## XtPopup

### FORMAT

XtPopup  
(popup\_shell, grab\_kind)

### Argument Information

| Argument    | Type       | Access | Mechanism |
|-------------|------------|--------|-----------|
| popup_shell | Widget     | read   | value     |
| grab_kind   | XtGrabKind | read   | value     |

---

## XtPopupSpringLoaded

### FORMAT

XtPopupSpringLoaded  
(popup\_shell)

### Argument Information

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| popup_shell | Widget | read   | value     |

---

## XtProcessEvent

### FORMAT

XtProcessEvent  
(mask)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| mask     | XtInputMask | read   | value     |

This routine, superseded by XtAppProcessEvent, is supported for backward compatibility.

---

## XtQueryGeometry

### FORMAT

status XtQueryGeometry  
(w, intended, preferred\_return)

### Argument Information

| Argument         | Type             | Access | Mechanism |
|------------------|------------------|--------|-----------|
| status           | XtGeometryResult | write  | value     |
| w                | Widget           | read   | value     |
| intended         | XtWidgetGeometry | write  | value     |
| preferred_return | XtWidgetGeometry | write  | reference |

---

## XtRealizeWidget

### FORMAT

XtRealizeWidget  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XtRealloc

### FORMAT

result XtRealloc  
(ptr, num)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | ADDRESS | write  | value     |
| ptr      | ADDRESS | read   | value     |

## Intrinsics Routines

### XtRealloc

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| num      | Cardinal | read   | value     |

---

## XtRegisterCaseConverter

### FORMAT

XtRegisterCaseConverter  
(display, proc, start, stop)

### Argument Information

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| display  | Display    | read   | value     |
| proc     | XtCaseProc | read   | value     |
| start    | KeySym     | read   | value     |
| stop     | KeySym     | read   | value     |

---

## XtRegisterGrabAction

### FORMAT

XtRegisterGrabAction  
(action\_proc, owner\_events, event\_mask, pointer\_mode, keyboard\_mode )

### Argument Information

| Argument      | Type              | Access | Mechanism |
|---------------|-------------------|--------|-----------|
| action_proc   | XtActionProc      | read   | value     |
| owner_events  | Boolean           | read   | value     |
| event_mask    | LONGWORD UNSIGNED | read   | value     |
| pointer_mode  | LONGWORD          | read   | value     |
| keyboard_mode | LONGWORD          | read   | value     |

---

## XtReleaseGC

### FORMAT

XtReleaseGC  
(object, gc)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| object   | Widget | read   | value     |
| gc       | GC     | read   | value     |

---

## XtRemoveActionHook

### FORMAT

XtRemoveActionHook  
(id)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| id       | XtActionHookId | read   | value     |

---

## XtRemoveAllCallbacks

### FORMAT

XtRemoveAllCallbacks  
(w, callback\_name)

### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| w             | Widget   | read   | value     |
| callback_name | XtString | read   | reference |

## Intrinsics Routines

### XtRemoveCallback

---

## XtRemoveCallback

### FORMAT

XtRemoveCallback  
(w, callback\_name, callback, client\_data)

### Argument Information

| Argument      | Type           | Access | Mechanism |
|---------------|----------------|--------|-----------|
| w             | Widget         | read   | value     |
| callback_name | XtString       | read   | reference |
| callback      | XtCallbackProc | read   | value     |
| client_data   | XtPointer      | read   | value     |

---

## XtRemoveCallbacks

### FORMAT

XtRemoveCallbacks  
(w, callback\_name, callbacks)

### Argument Information

| Argument      | Type                  | Access | Mechanism |
|---------------|-----------------------|--------|-----------|
| w             | Widget                | read   | value     |
| callback_name | XtString              | read   | reference |
| callbacks     | XtCallbackRec (array) | read   | reference |

---

## XtRemoveEventHandler

### FORMAT

XtRemoveEventHanlder  
(w, event\_mask, nonmaskable, proc, client\_data)



Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| w           | Widget         | read   | value     |
| event_mask  | EventMask      | read   | value     |
| nonmaskable | Boolean        | read   | value     |
| proc        | XtEventHandler | read   | reference |
| client_data | XtPointer      | read   | value     |

---

**XtRemoveGrab**

**FORMAT**

XtRemoveGrab  
(w)

Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

**XtRemoveInput**

**FORMAT**

XtRemoveInput  
(id)

Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| id       | XtInputId | read   | value     |

## Intrinsics Routines

### XtRemoveRawEventHandler

---

## XtRemoveRawEventHandler

### FORMAT

XtRemoveRawEventHandler  
(w, event\_mask, nonmaskable, proc, client\_data)

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| w           | Widget         | read   | value     |
| event_mask  | EventMask      | read   | value     |
| nonmaskable | Boolean        | read   | value     |
| proc        | XtEventHandler | read   | value     |
| client_data | XtPointer      | read   | value     |

---

## XtRemoveTimeOut

### FORMAT

XtRemoveTimeOut  
(timer)

### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| timer    | XtIntervalId | read   | value     |

---

## XtRemoveWorkProc

### FORMAT

XtRemoveWorkProc  
(id)

### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| id       | XtWorkProcId | read   | value     |

## XtResolvePathname

### FORMAT

result XtResolvePathname  
(dpy, type, filename, suffix, path, substitutions, num\_substitutions, predicate)

### Argument Information

| Argument          | Type            | Access | Mechanism |
|-------------------|-----------------|--------|-----------|
| result            | XtString        | write  | value     |
| dpy               | Display         | read   | value     |
| type              | XtString        | read   | reference |
| filename          | XtString        | read   | reference |
| suffix            | XtString        | read   | reference |
| path              | XtString        | read   | reference |
| substitutions     | Substitution    | read   | reference |
| num_substitutions | Cardinal        | read   | value     |
| predicate         | XtFilePredicate | read   | value     |

## XtScreen

### FORMAT

result XtScreen  
(w)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | ADDRESS | write  | value     |
| w        | Widget  | read   | value     |

## Intrinsics Routines

### XtScreenDatabase

---

## XtScreenDatabase

### FORMAT

result XtScreenDatabase  
(screen)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | XrmDatabase | write  | value     |
| screen   | Screen      | read   | value     |

---

## XtScreenOfObject

### FORMAT

result XtScreenOfObject  
(object)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | ADDRESS | write  | value     |
| object   | Widget  | read   | value     |

---

## XtSetErrorHandler

### FORMAT

XtSetErrorHandler  
(handler)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| handler  | XtErrorHandler | read   | value     |

This routine, superseded by XtAppSetErrorHandler, is supported for backward compatibility.

## XtSetErrorMsgHandler

### FORMAT

XtSetErrorMsgHandler  
(msg\_handler)

### Argument Information

| Argument    | Type              | Access | Mechanism |
|-------------|-------------------|--------|-----------|
| msg_handler | XtErrorMsgHandler | read   | value     |

This routine, superseded by XtAppSetWarningHandler, is supported for backward compatibility.

## XtSetKeyboardFocus

### FORMAT

XtSetKeyboardFocus  
(subtree, descendant)

### Argument Information

| Argument   | Type   | Access | Mechanism |
|------------|--------|--------|-----------|
| subtree    | Widget | read   | value     |
| descendant | Widget | read   | value     |

## XtSetKeyTranslator

### FORMAT

XtSetKeyTranslator  
(display, proc)

### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| display  | Display   | read   | value     |
| proc     | XtKeyProc | read   | value     |

## Intrinsics Routines

### XtSetLanguageProc

---

## XtSetLanguageProc

### FORMAT

result XtSetLanguageProc  
(app\_context, proc, client\_data)

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| result      | XtLanguageProc | write  | value     |
| app_context | XtAppContext   | read   | value     |
| proc        | XtLanguageProc | read   | value     |
| client_data | XtPointer      | read   | value     |

---

## XtSetMappedWhenManaged

### FORMAT

XtSetMappedWhenManaged  
(w, mapped\_when\_managed)

### Argument Information

| Argument            | Type    | Access | Mechanism |
|---------------------|---------|--------|-----------|
| w                   | Widget  | read   | value     |
| mapped_when_managed | Boolean | read   | value     |

---

## XtSetMultiClickTime

### FORMAT

XtSetMultiClickTime  
(display, time)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| display  | Display  | read   | value     |
| time     | LONGWORD | read   | value     |

**XtSetSelectionTimeout**

**FORMAT**

XtSetSelectionTimeout  
(timeout)

**Argument Information**

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| timeout  | LONGWORD UNSIGNED | read   | value     |

This routine, superseded by XtAppSetSelectionTimeout, is supported for backward compatibility.

**XtSetSensitive**

**FORMAT**

XtSetSensitive  
(w, sensitive)

**Argument Information**

| Argument  | Type    | Access | Mechanism |
|-----------|---------|--------|-----------|
| w         | Widget  | read   | value     |
| sensitive | Boolean | read   | value     |

## Intrinsics Routines

### XtSetSubvalues

---

## XtSetSubvalues

### FORMAT

XtSetSubvalues

(base, resources, num\_resources, args, num\_args)

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| base          | XtPointer          | read   | value     |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| args          | Arg (array)        | read   | reference |
| num_args      | Cardinal           | read   | value     |

---

## XtSetTypeConverter

### FORMAT

XtSetTypeConverter

(from\_type, toType, converter, convert\_args, num\_args, cache\_type, destructor)

### Argument Information

| Argument     | Type                    | Access | Mechanism |
|--------------|-------------------------|--------|-----------|
| from_type    | XtString                | read   | reference |
| toType       | XtString                | read   | reference |
| converter    | XtTypeConverter         | read   | value     |
| convert_args | XtConvertArgRec (array) | read   | reference |
| num_args     | Cardinal                | read   | value     |
| cache_type   | XtCacheType             | read   | value     |
| destructor   | XtDestructor            | read   | value     |



---

## XtSetValues

### FORMAT

XtSetValues  
(object, args, num\_args)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| object   | Widget      | read   | value     |
| args     | Arg (array) | read   | reference |
| num_args | Cardinal    | read   | value     |

---

## XtSetWarningHandler

### FORMAT

XtSetWarningHandler  
(handler)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| handler  | XtErrorHandler | read   | value     |

This routine, superseded by XtAppSetWarningHandler, is supported for backward compatibility.

---

## XtSetWarningMsgHandler

### FORMAT

XtSetWarningMsgHandler  
(msg\_handler)

### Argument Information

| Argument    | Type              | Access | Mechanism |
|-------------|-------------------|--------|-----------|
| msg_handler | XtErrorMsgHandler | read   | value     |

## Intrinsics Routines

### XtSetWarningMsgHandler

This routine, superseded by XtSetWarningMsgHandler, is supported for backward compatibility.

---

## XtSetWMColormapWindows

### FORMAT

XtSetWMColormapWindows  
(widget, widget\_list, count)

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| widget      | Widget         | read   | value     |
| widget_list | Widget (array) | read   | reference |
| count       | Cardinal       | read   | value     |

---

## XtStringConversionWarning

### FORMAT

XtStringConversionWarning  
(from, toType )

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| from     | XtString | read   | value     |
| toType   | XtString | read   | value     |

This procedure is a convenience routine for use by prior to R4 resource converters that convert from strings to other formats.

---

## XtSuperclass

### FORMAT

result XtSuperclass  
(w)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | WidgetClass | write  | value     |
| w        | Widget      | read   | value     |

---

**XtToolkitInitialize**

**FORMAT**

XtToolkitInitialize  
( )

---

**XtTranslateCoords**

**FORMAT**

XtTranslateCoords  
(w, x, y, rootx\_return, rooty\_return)

**Argument Information**

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| w            | Widget   | read   | value     |
| x            | Position | read   | value     |
| y            | Position | read   | value     |
| rootx_return | Position | write  | reference |
| rooty_return | Position | write  | reference |

---

**XtTranslateKey**

**FORMAT**

XtTranslateKey  
(display, keycode, modifiers, modifiers\_return, keysym\_return)

## Intrinsics Routines

### XtTranslateKey

#### Argument Information

| Argument         | Type      | Access | Mechanism |
|------------------|-----------|--------|-----------|
| display          | Display   | read   | value     |
| keycode          | KeyCode   | read   | value     |
| modifiers        | Modifiers | read   | value     |
| modifiers_return | Modifiers | write  | reference |
| keysym_return    | KeySym    | write  | reference |

---

## XtTranslateKeycode

### FORMAT

XtTranslateKeycode

(display, keycode, modifiers, modifiers\_return, keysym\_return)

#### Argument Information

| Argument         | Type      | Access | Mechanism |
|------------------|-----------|--------|-----------|
| display          | Display   | read   | value     |
| keycode          | KeyCode   | read   | value     |
| modifiers        | Modifiers | read   | value     |
| modifiers_return | Modifiers | write  | reference |
| keysym_return    | KeySym    | write  | reference |

---

## XtUngrabButton

### FORMAT

XtUngrabButton

(widget, button, modifiers)

#### Argument Information

| Argument  | Type     | Access | Mechanism |
|-----------|----------|--------|-----------|
| widget    | Widget   | read   | value     |
| button    | LONGWORD | read   | value     |
| modifiers | Modifier | read   | value     |

---

## XtUngrabKey

### FORMAT

XtUngrabKey  
(widget, keycode, modifiers)

### Argument Information

| Argument  | Type      | Access | Mechanism |
|-----------|-----------|--------|-----------|
| widget    | Widget    | read   | value     |
| keycode   | KeyCode   | read   | value     |
| modifiers | Modifiers | read   | value     |

---

## XtUngrabKeyboard

### FORMAT

XtUngrabKeyboard  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

---

## XtUngrabPointer

### FORMAT

XtUngrabPointer  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

## Intrinsics Routines

### XtUninstallTranslations

---

## XtUninstallTranslations

### FORMAT

XtUninstallTranslations  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XtUnmanageChild

### FORMAT

XtUnmanageChild  
(child)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| child    | Widget | read   | value     |

---

## XtUnmanageChildren

### FORMAT

XtUnmanageChildren  
(children, num\_children)

### Argument Information

| Argument     | Type           | Access | Mechanism |
|--------------|----------------|--------|-----------|
| children     | Widget (array) | read   | reference |
| num_children | Cardinal       | read   | value     |

---

## XtUnrealizeWidget

### FORMAT

XtUnrealizeWidget  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XtVaAppCreateShell

### FORMAT

XtVaAppCreateShell  
(application\_name, application\_class, widget\_class, display, [name, value]...)

### Argument Information

| Argument          | Type        | Access | Mechanism |
|-------------------|-------------|--------|-----------|
| application_name  | XtString    | read   | reference |
| application_class | XtString    | read   | reference |
| widget_class      | WidgetClass | read   | value     |
| display           | Display     | read   | value     |
| name              | XtString    | read   | reference |
| value             | XtArgVal    | read   | value     |

---

## XtVaAppInitialize

### FORMAT

XtVaAppInitialize  
(app\_context\_return, application\_class, options, num\_options, argc\_in\_out, argv\_in\_out, fallback\_resources, [name, value]...)

## Intrinsics Routines

### XtVaAppInitialize

#### Argument Information

| Argument           | Type                         | Access | Mechanism |
|--------------------|------------------------------|--------|-----------|
| app_context_return | XtAppContext                 | write  | reference |
| application_class  | XtString                     | read   | reference |
| options            | LONGWORD UNSIGNED<br>(array) | read   | reference |
| num_options        | Cardinal                     | read   | value     |
| argc_in_out        | Cardinal                     | read   | value     |
| argv_in_out        | XtString (array)             | write  | reference |
| fallback_resources | XtString (array)             | write  | reference |
| name               | XtString                     | read   | reference |
| value              | XtArgVal                     | read   | value     |

---

## XtVaCreateArgsList

### FORMAT

result XtVaCreateArgsList  
(unused, [name, value]...)

#### Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | XtVarArgsList | write  | value     |
| unused   | XtPointer     | read   | value     |
| name     | XtString      | read   | reference |
| value    | XtArgVal      | read   | value     |

---

## XtVaCreateManagedWidget

### FORMAT

result XtVaCreateManagedWidget  
(name, widget\_class, parent, [name, value]...)



**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | Widget      | write         | value            |
| name            | XtString    | read          | reference        |
| widget_class    | WidgetClass | read          | value            |
| parent          | Widget      | read          | value            |
| name            | XtString    | read          | reference        |
| value           | XtArgVal    | read          | value            |

**XtVaCreatePopupShell**

**FORMAT**

```
XtVaCreatePopupShell
    (name, widget_class, parent, [name, value]...)
```

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| name            | XtString    | read          | reference        |
| widget_class    | WidgetClass | read          | value            |
| parent          | Widget      | read          | value            |
| name            | XtString    | read          | reference        |
| value           | XtArgVal    | read          | value            |

**XtVaCreateWidget**

**FORMAT**

```
result XtVaCreateWidget
    (name, object_class, parent, [name, value]...)
```

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | Widget      | write         | value            |
| name            | XtString    | read          | reference        |
| object_class    | WidgetClass | read          | value            |

## Intrinsics Routines

### XtVaCreateWidget

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| value    | XtArgVal | read   | value     |

---

## XtVaGetApplicationResources

### FORMAT

XtVaGetApplicationResources

(w, base, resources, num\_resources, [name, value]...)

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| w             | Widget             | read   | value     |
| base          | XtPointer          | read   | value     |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| name          | XtString           | read   | reference |
| value         | XtArgVal           | read   | value     |

---

## XtVaGetSubresources

### FORMAT

XtVaGetSubresources

(w, base, name, class, resources, num\_resources, [name, value]...)

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| w             | Widget             | read   | value     |
| base          | XtPointer          | read   | value     |
| name          | XtString           | read   | reference |
| class         | XtString           | read   | reference |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| name          | XtString           | read   | reference |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| value    | XtArgVal | read   | value     |

---

## XtVaGetSubvalues

### FORMAT

XtVaGetSubvalues  
(base, resources, num\_resources, [name, value...])

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| base          | XtPointer          | read   | value     |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| name          | XtString           | read   | reference |
| value         | XtArgVal           | read   | value     |

---

## XtVaGetValues

### FORMAT

XtVaGetValues  
(object, [name, value]...)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| object   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| value    | XtArgVal | read   | reference |

## Intrinsics Routines

### XtVaSetSubvalues

---

## XtVaSetSubvalues

### FORMAT

XtVaSetSubvalues

(base, resources, num\_resources, [name, value]...)

### Argument Information

| Argument      | Type               | Access | Mechanism |
|---------------|--------------------|--------|-----------|
| base          | XtPointer          | read   | value     |
| resources     | XtResource (array) | read   | reference |
| num_resources | Cardinal           | read   | value     |
| name          | XtString           | read   | reference |
| value         | XtArgVal           | read   | value     |

---

## XtVaSetValues

### FORMAT

XtVaSetValues

(object, [name, value]...)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| object   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| value    | XtArgVal | read   | value     |

---

## XtWarning

### FORMAT

XtWarning

(message)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| message  | XtString | read   | value     |

This routine, superseded by XtAppWarningMsg, is supported for backward compatibility.

**XtWarningMsg**

**FORMAT**

XtWarningMsg  
(name, type, class, default, params, num\_params)

**Argument Information**

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| name       | XtString         | read   | reference |
| type       | XtString         | read   | reference |
| class      | XtString         | read   | reference |
| default    | XtString         | read   | reference |
| params     | XtString (array) | read   | reference |
| num_params | Cardinal         | read   | reference |

This routine, superseded by XtAppWarningMsg, is supported for backward compatibility.

**XtWidgetToApplicationContext**

**FORMAT**

result XtWidgetToApplicationContext  
(w)

**Argument Information**

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| result   | XtAppContext | write  | value     |
| w        | Widget       | read   | value     |

## Intrinsics Routines

### XtWindow

---

## XtWindow

### FORMAT

result XtWindow  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Window | write  | value     |
| w        | Widget | read   | value     |

---

## XtWindowOfObject

### FORMAT

result XtWindowOfObject  
(object)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Window | write  | value     |
| object   | Widget | read   | value     |

---

## XtWindowToWidget

### FORMAT

widget XtWindowToWidget  
(display, window)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | write  | value     |
| display  | Display | read   | value     |
| window   | Window  | read   | value     |

---

---

## Intrinsics Data Structures

This chapter documents intrinsics data structures.

---

### Arg

#### Field Information

---

| Argument | Type     |
|----------|----------|
| name     | ADDRESS  |
| value    | XtArgVal |

---

---

### XtActionsRec

#### Field Information

---

| Argument | Type         |
|----------|--------------|
| strng    | XtString     |
| proc     | XtActionProc |

---

---

### XtCallbackRec

#### Field Information

---

| Argument | Type           |
|----------|----------------|
| callback | XtCallbackProc |
| closure  | XtPointer      |

---

## Intrinsics Data Structures

### XtConvertArgRec

---

## XtConvertArgRec

### Field Information

---

| Argument     | Type          |
|--------------|---------------|
| address_mode | XtAddressMode |
| address_id   | XtPointer     |
| size         | Cardinal      |

---

---

## XtI18nContextRec

### Field Information

---

| Argument          | Type    |
|-------------------|---------|
| locale            | ADDRESS |
| use_mrm_hierarchy | Boolean |
| mrm_hierarchy_id  | Opaque  |
| widget_class      | Opaque  |

---

---

## XtPopdownIdRec

### Field Information

---

| Argument      | Type   |
|---------------|--------|
| shell_widget  | Widget |
| enable_widget | Widget |

---

---

## XtResource

### Field Information

---

| Argument       | Type     |
|----------------|----------|
| resource_name  | XtString |
| resource_class | XtString |

---



---

| Argument        | Type      |
|-----------------|-----------|
| resource_type   | XtString  |
| resource_size   | Cardinal  |
| resource_offset | Cardinal  |
| default_type    | XtString  |
| default_addr    | XtPointer |

---

---

## XtSubstitutionRec

### Field Information

---

| Argument     | Type     |
|--------------|----------|
| match        | BYTE     |
| substitution | XtString |

---

---

## XtWidgetGeometry

### Field Information

---

| Argument     | Type           |
|--------------|----------------|
| request_mode | XtGeometryMask |
| x            | Position       |
| y            | Position       |
| width        | Dimension      |
| height       | Dimension      |
| border_width | Dimension      |
| sibling      | Widget         |
| stack_mode   | LONGWORD       |

---



# Part III

---

## OSF/Motif Toolkit

Part III documents OSF/Motif routines and data structures and includes the following chapters:

- Chapter 6 – OSF/Motif Toolkit Routines
- Chapter 7 – OSF/Motif Toolkit Data Structures



---

## OSF/Motif Toolkit Routines

This chapter documents OSF/Motif Toolkit routines. See Section 1.2 for information about the format used to describe each routine.

---

### MrmCloseHierarchy

#### FORMAT

```
result MrmCloseHierarchy
    (hierarchy_id)
```

#### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |

---

### MrmFetchBitmapLiteral

#### FORMAT

```
result MrmFetchBitmapLiteral
    (hierarchy_id, index, screen, display, pixmap, width, height)
```

#### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |
| index        | XtString     | read   | reference |
| screen       | Screen       | read   | reference |
| display      | Display      | read   | value     |
| pixmap       | Pixmap       | write  | reference |
| width        | Dimension    | write  | reference |

## OSF/Motif Toolkit Routines

### MrmFetchBitmapLiteral

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| height   | Dimension | write  | reference |

---

### MrmFetchColorLiteral

#### FORMAT

result MrmFetchColorLiteral  
(hierarchy\_id, index, display, colormap\_id, pixel)

#### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |
| index        | XtString     | read   | reference |
| display      | Display      | read   | value     |
| colormap_id  | Colormap     | read   | value     |
| pixel        | Pixel        | write  | reference |

---

### MrmFetchIconLiteral

#### FORMAT

result MrmFetchIconLiteral  
(hierarchy\_id, index, screen, display, fgpix, bgpix, pixmap)

#### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |
| index        | XtString     | read   | reference |
| screen       | Screen       | read   | reference |
| display      | Display      | read   | value     |
| fgpix        | Pixel        | read   | value     |
| bgpix        | Pixel        | read   | value     |
| pixmap       | Pixmap       | write  | reference |

---

## MrmFetchInterfaceModule

### FORMAT

result MrmFetchInterfaceModule  
(hierarchy\_id, module\_name, parent, w\_return)

### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |
| module_name  | XtString     | read   | reference |
| parent       | Widget       | read   | value     |
| w_return     | Widget       | write  | reference |

---

## MrmFetchLiteral

### FORMAT

result MrmFetchLiteral  
(hierarchy\_id, index, display, value, type)

### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |
| index        | XtString     | read   | reference |
| display      | Display      | read   | value     |
| value        | XtPointer    | write  | reference |
| type         | MrmCode      | write  | reference |

---

## MrmFetchSetValues

### FORMAT

result MrmFetchSetValues  
(hierarchy\_id, widget, args, num\_args)

### Argument Information

| Argument     | Type         | Access | Mechanism |
|--------------|--------------|--------|-----------|
| result       | Cardinal     | write  | value     |
| hierarchy_id | MrmHierarchy | read   | value     |
| widget       | Widget       | read   | value     |
| args         | Arg (array)  | read   | reference |
| num_args     | Cardinal     | read   | value     |

---

## MrmFetchWidget

### FORMAT

result MrmFetchWidget  
(hierarchy\_id, index, parent\_widget, widget, class)

### Argument Information

| Argument      | Type         | Access | Mechanism |
|---------------|--------------|--------|-----------|
| result        | Cardinal     | write  | value     |
| hierarchy_id  | MrmHierarchy | read   | value     |
| index         | XtString     | read   | reference |
| parent_widget | Widget       | read   | value     |
| widget        | Widget       | write  | reference |
| class         | MrmType      | write  | reference |



---

## MrmFetchWidgetOverride

### FORMAT

result MrmFetchWidgetOverride  
(hierarchy\_id, index, parent\_widget, override\_name, override\_args,  
override\_num\_args, widget, class)

### Argument Information

| Argument          | Type         | Access | Mechanism |
|-------------------|--------------|--------|-----------|
| result            | Cardinal     | write  | value     |
| hierarchy_id      | MrmHierarchy | read   | value     |
| index             | XtString     | read   | reference |
| parent_widget     | Widget       | read   | value     |
| override_name     | XtString     | read   | reference |
| override_args     | Arg (array)  | read   | reference |
| override_num_args | Cardinal     | read   | value     |
| widget            | Widget       | write  | reference |
| class             | MrmType      | write  | reference |

---

## MrmInitialize

### FORMAT

MrmInitialize

---

## MrmOpenHierarchy

### FORMAT

result MrmOpenHierarchy  
(num\_files, file\_names\_list, ancillary\_structures\_list, hierarchy\_id\_return)

### Argument Information

| Argument        | Type             | Access | Mechanism |
|-----------------|------------------|--------|-----------|
| result          | Cardinal         | write  | value     |
| num_files       | MrmCount         | read   | value     |
| file_names_list | XtString (array) | read   | reference |

## OSF/Motif Toolkit Routines

### MrmOpenHierarchy

---

| Argument                  | Type             | Access | Mechanism |
|---------------------------|------------------|--------|-----------|
| ancillary_structures_list | XtString (array) | read   | reference |
| hierarchy_id_return       | MrmHierarchy     | write  | reference |

---

### MrmOpenHierarchyPerDisplay

#### FORMAT

result MrmOpenHierarchyPerDisplay  
(display, num\_files, file\_names\_list, ancillary\_structures\_list, hierarchy\_id\_return)

#### Argument Information

---

| Argument                  | Type             | Access | Mechanism |
|---------------------------|------------------|--------|-----------|
| result                    | Cardinal         | write  | value     |
| display                   | Display          | read   | value     |
| num_files                 | MrmCount         | read   | value     |
| file_names_list           | XtString (array) | read   | reference |
| ancillary_structures_list | XtString (array) | read   | reference |
| hierarchy_id_return       | MrmHierarchy     | write  | reference |

---

### MrmRegisterClass

#### FORMAT

result MrmRegisterClass  
(class\_code, class\_name, create\_name, create\_proc, class\_record)

#### Argument Information

---

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| result       | Cardinal    | write  | value     |
| class_code   | MrmType     | read   | value     |
| class_name   | XtString    | read   | reference |
| create_name  | XtString    | read   | reference |
| create_proc  | WidgetProc  | read   | value     |
| class_record | WidgetClass | read   | value     |

---

---

## MrmRegisterNames

### FORMAT

result MrmRegisterNames  
(register\_list, register\_count)

### Argument Information

| Argument       | Type                      | Access | Mechanism |
|----------------|---------------------------|--------|-----------|
| result         | Cardinal                  | write  | value     |
| register_list  | MrmRegisterArg<br>(array) | read   | reference |
| register_count | MrmCount                  | read   | value     |

---

## MrmRegisterNamesInHierarchy

### FORMAT

MrmRegisterNamesInHierarchy  
(hierarchy\_id, register\_list, register\_count)

### Argument Information

| Argument       | Type                      | Access | Mechanism |
|----------------|---------------------------|--------|-----------|
| hierarchy_id   | MrmHierarchy              | read   | value     |
| register_list  | MrmRegisterArg<br>(array) | read   | reference |
| register_count | MrmCount                  | read   | value     |

---

## XmActivateProtocol

### FORMAT

XmActivateProtocol  
(shell, property, protocol)

## OSF/Motif Toolkit Routines

### XmActivateProtocol

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| shell    | Widget | read   | value     |
| property | Atom   | read   | value     |
| protocol | Atom   | read   | value     |

---

## XmAddProtocolCallback

### FORMAT

XmAddProtocolCallback  
(shell, property, protocol, callback, closure)

#### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| shell    | Widget         | read   | value     |
| property | Atom           | read   | value     |
| protocol | Atom           | read   | value     |
| callback | XtCallbackProc | read   | value     |
| closure  | XtPointer      | read   | value     |

---

## XmAddProtocols

### FORMAT

XmAddProtocols  
(shell, property, protocols, num\_protocols)

#### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| shell         | Widget   | read   | value     |
| property      | Atom     | read   | value     |
| protocols     | Atom     | read   | reference |
| num_protocols | Cardinal | read   | value     |

---

## XmAddTabGroup

### FORMAT

XmAddTabGroup  
(tab\_group)

### Argument Information

| Argument  | Type   | Access | Mechanism |
|-----------|--------|--------|-----------|
| tab_group | Widget | read   | value     |

---

## XmAddToPostFromList

### FORMAT

XmAddToPostFromList  
(menu\_wid, widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| menu_wid | Widget | read   | value     |
| widget   | Widget | read   | value     |

---

## XmCascadeButtonGadgetHighlight

### FORMAT

XmCascadeButtonGadgetHighlight  
(cascadeButtonGadget, highlight)

### Argument Information

| Argument            | Type    | Access | Mechanism |
|---------------------|---------|--------|-----------|
| cascadeButtonGadget | Widget  | read   | value     |
| highlight           | Boolean | read   | value     |

---

## XmCascadeButtonHighlight

### FORMAT

XmCascadeButtonHighlight  
(cb, highlight)

### Argument Information

| Argument  | Type    | Access | Mechanism |
|-----------|---------|--------|-----------|
| cb        | Widget  | read   | value     |
| highlight | Boolean | read   | value     |

---

## XmChangeColor

### FORMAT

XmChangeColor  
(widget, background)

### Argument Information

| Argument   | Type   | Access | Mechanism |
|------------|--------|--------|-----------|
| widget     | Widget | read   | value     |
| background | Pixel  | read   | value     |

---

## XmClipboardBeginCopy

### FORMAT

result XmClipboardBeginCopy  
(display, window, label, widget, callback, itemid)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |
| window   | Window   | read   | value     |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| label    | XmString | read   | value     |
| widget   | Widget   | read   | value     |
| callback | VoidProc | read   | value     |
| itemid   | LONGWORD | read   | reference |

---

## XmClipboardCancelCopy

### FORMAT

result XmClipboardCancelCopy  
(display, window, item\_id)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |
| window   | Window   | read   | value     |
| item_id  | LONGWORD | read   | value     |

---

## XmClipboardCopy

### FORMAT

result XmClipboardCopy  
(display, window, item\_id, format\_name, buffer, length, private\_id, data\_id)

### Argument Information

| Argument    | Type                 | Access | Mechanism |
|-------------|----------------------|--------|-----------|
| result      | LONGWORD             | write  | value     |
| display     | Display              | read   | value     |
| window      | Window               | read   | value     |
| item_id     | LONGWORD             | read   | value     |
| format_name | XtString             | read   | reference |
| buffer      | XtPointer            | read   | value     |
| length      | LONGWORD<br>UNSIGNED | read   | value     |

## OSF/Motif Toolkit Routines

### XmClipboardCopy

---

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| private_id | LONGWORD | read   | value     |
| data_id    | LONGWORD | modify | reference |

---

### XmClipboardCopyByName

#### FORMAT

result XmClipboardCopyByName  
(display, window, data\_id, buffer, length, private\_id)

#### Argument Information

---

| Argument   | Type                 | Access | Mechanism |
|------------|----------------------|--------|-----------|
| result     | LONGWORD             | write  | value     |
| display    | Display              | read   | value     |
| window     | Window               | read   | value     |
| data_id    | LONGWORD             | read   | value     |
| buffer     | XtPointer            | read   | value     |
| length     | LONGWORD<br>UNSIGNED | read   | value     |
| private_id | LONGWORD             | read   | value     |

---

### XmClipboardEndCopy

#### FORMAT

result XmClipboardEndCopy  
(display, window, item\_id)

#### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |
| window   | Window   | read   | value     |
| item_id  | LONGWORD | read   | value     |

---



---

## XmClipboardEndRetrieve

### FORMAT

result XmClipboardEndRetrieve  
(display, window)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |
| window   | Window   | read   | value     |

---

## XmClipboardInquireCount

### FORMAT

result XmClipboardInquireCount  
(display, window, count, max\_format\_name\_length)

### Argument Information

| Argument                   | Type               | Access | Mechanism |
|----------------------------|--------------------|--------|-----------|
| result                     | LONGWORD           | write  | value     |
| display                    | Display            | read   | value     |
| window                     | Window             | read   | value     |
| count                      | LONGWORD           | write  | reference |
| max_format_name_<br>length | LONGWORD<br>SIGNED | write  | reference |

---

## XmClipboardInquireFormat

### FORMAT

result XmClipboardInquireFormat  
(display, window, index, format\_name\_buf, buffer\_len, copied\_len)

## OSF/Motif Toolkit Routines

### XmClipboardInquireFormat

#### Argument Information

| Argument        | Type                 | Access | Mechanism |
|-----------------|----------------------|--------|-----------|
| result          | LONGWORD             | write  | value     |
| display         | Display              | read   | value     |
| window          | Window               | read   | value     |
| index           | LONGWORD             | read   | value     |
| format_name_buf | XtPointer            | read   | value     |
| buffer_len      | LONGWORD<br>UNSIGNED | read   | value     |
| copied_len      | LONGWORD<br>UNSIGNED | write  | reference |

---

### XmClipboardInquireLength

#### FORMAT

result XmClipboardInquireLength  
(display, window, format\_name, length)

#### Argument Information

| Argument    | Type                 | Access | Mechanism |
|-------------|----------------------|--------|-----------|
| result      | LONGWORD             | write  | value     |
| display     | Display              | read   | value     |
| window      | Window               | read   | value     |
| format_name | XtString             | read   | reference |
| length      | LONGWORD<br>UNSIGNED | write  | reference |

---

### XmClipboardInquirePendingItems

#### FORMAT

result XmClipboardInquirePendingItems  
(display, window, format\_name, item\_list, count)

**Argument Information**

| <b>Argument</b> | <b>Type</b>            | <b>Access</b> | <b>Mechanism</b> |
|-----------------|------------------------|---------------|------------------|
| result          | LONGWORD               | write         | value            |
| display         | Display                | read          | value            |
| window          | Window                 | read          | value            |
| format_name     | XtString               | read          | reference        |
| item_list       | XmClipboardPendingList | write         | reference        |
| count           | LONGWORD UNSIGNED      | write         | reference        |

---

**XmClipboardLock**

**FORMAT**

```
result XmClipboardLock
    (display, window)
```

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | LONGWORD    | write         | value            |
| display         | Display     | read          | value            |
| window          | Window      | read          | value            |

---

**XmClipboardRegisterFormat**

**FORMAT**

```
result XmClipboardRegisterFormat
    (display, format_name, format_length)
```

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | LONGWORD    | write         | value            |
| display         | Display     | read          | value            |
| format_name     | XtString    | read          | reference        |
| format_length   | LONGWORD    | read          | value            |

---

## XmClipboardRetrieve

### FORMAT

result XmClipboardRetrieve  
(display, window, format\_name, buffer, length, num\_bytes, private\_id)

### Argument Information

| Argument    | Type                 | Access | Mechanism |
|-------------|----------------------|--------|-----------|
| result      | LONGWORD             | write  | value     |
| display     | Display              | read   | value     |
| window      | Window               | read   | value     |
| format_name | XtString             | read   | reference |
| buffer      | XtPointer            | read   | value     |
| length      | LONGWORD<br>UNSIGNED | read   | value     |
| num_bytes   | LONGWORD<br>UNSIGNED | write  | reference |
| private_id  | LONGWORD             | modify | reference |

---

## XmClipboardStartCopy

### FORMAT

result XmClipboardStartCopy  
(display, window, clip\_label, timestamp, widget, callback, item\_id)

### Argument Information

| Argument   | Type           | Access | Mechanism |
|------------|----------------|--------|-----------|
| result     | LONGWORD       | write  | value     |
| display    | Display        | read   | value     |
| window     | Window         | read   | value     |
| clip_label | XmString       | read   | value     |
| timestamp  | Time           | read   | value     |
| widget     | Widget         | read   | value     |
| callback   | XmCutPasteProc | read   | value     |
| item_id    | LONGWORD       | read   | reference |

---

## XmClipboardStartRetrieve

### FORMAT

result XmClipboardStartRetrieve  
(display, window, timestamp )

### Argument Information

| Argument  | Type     | Access | Mechanism |
|-----------|----------|--------|-----------|
| result    | LONGWORD | write  | value     |
| display   | Display  | read   | value     |
| window    | Window   | read   | value     |
| timestamp | Time     | read   | value     |

---

## XmClipboardUndoCopy

### FORMAT

result XmClipboardUndoCopy  
(display, window)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |
| window   | Window   | read   | value     |

---

## XmClipboardUnlock

### FORMAT

result XmClipboardUnlock  
(display, window, remove\_all\_locks)

## OSF/Motif Toolkit Routines

### XmClipboardUnlock

#### Argument Information

| Argument         | Type     | Access | Mechanism |
|------------------|----------|--------|-----------|
| result           | LONGWORD | write  | value     |
| display          | Display  | read   | value     |
| window           | Window   | read   | value     |
| remove_all_locks | Boolean  | read   | value     |

---

## XmClipboardWithdrawFormat

### FORMAT

result XmClipboardWithdrawFormat  
(display, window, data)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| display  | Display  | read   | value     |
| window   | Window   | read   | value     |
| data     | LONGWORD | read   | value     |

---

## XmCommandAppendValue

### FORMAT

XmCommandAppendValue  
(widget, command)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| command  | XmString | read   | value     |

---

## XmCommandError

### FORMAT

XmCommandError  
(widget, error)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| error    | XmString | read   | value     |

---

## XmCommandGetChild

### FORMAT

result XmCommandGetChild  
(widget, child)

### Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | Widget        | write  | value     |
| widget   | Widget        | read   | value     |
| child    | BYTE UNSIGNED | read   | value     |

---

## XmCommandSetValue

### FORMAT

XmCommandSetValue  
(widget, command)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| command  | XmString | read   | value     |

---

## XmConvertUnits

### FORMAT

result XmConvertUnits

(widget, orientation, from\_unit\_type, from\_value, to\_unit\_type)

### Argument Information

| Argument       | Type     | Access | Mechanism |
|----------------|----------|--------|-----------|
| result         | LONGWORD | write  | value     |
| widget         | Widget   | read   | value     |
| orientation    | LONGWORD | read   | value     |
| from_unit_type | LONGWORD | read   | value     |
| from_value     | LONGWORD | read   | value     |
| to_unit_type   | LONGWORD | read   | value     |

---

## XmCreateArrowButton

### FORMAT

widget XmCreateArrowButton

(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |



---

## XmCreateArrowButtonGadget

### FORMAT

widget XmCreateArrowButtonGadget  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateBulletinBoard

### FORMAT

widget XmCreateBulletinBoard  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | read   | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateBulletinBoardDialog

### FORMAT

widget XmCreateBulletinBoardDialog  
(parent, name, arglist, argcount)

## OSF/Motif Toolkit Routines

### XmCreateBulletinBoardDialog

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateCascadeButton

#### FORMAT

widget XmCreateCascadeButton  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateCascadeButtonGadget

#### FORMAT

widget XmCreateCascadeButtonGadget  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

## XmCreateCommand

### FORMAT

widget XmCreateCommand  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateCommandDialog

### FORMAT

widget XmCreateCommandDialog  
(ds\_p, name, fsb\_args, fsb\_n)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| ds_p     | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| fsb_args | Arg (array) | read   | reference |
| fsb_n    | Cardinal    | read   | value     |

## XmCreateDialogShell

### FORMAT

widget XmCreateDialogShell  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateDragIcon

### FORMAT

widget XmCreateDragIcon  
(parent, name, argList, argCount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| argList  | Arg (array) | read   | reference |
| argCount | Cardinal    | read   | value     |

---

## XmCreateDrawingArea

### FORMAT

widget XmCreateDrawingArea  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**XmCreateDrawnButton**

**FORMAT**

widget XmCreateDrawnButton  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**XmCreateErrorDialog**

**FORMAT**

widget XmCreateErrorDialog  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

## OSF/Motif Toolkit Routines

### XmCreateErrorDialog

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

---

## XmCreateFileSelectionBox

### FORMAT

widget XmCreateFileSelectionBox  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateFileSelectionDialog

### FORMAT

widget XmCreateFileSelectionDialog  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateForm

### FORMAT

widget XmCreateForm  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateFormDialog

### FORMAT

widget XmCreateForm  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateFrame

### FORMAT

widget XmCreateFrame  
(parent, name, arglist, argcount)

## OSF/Motif Toolkit Routines

### XmCreateFrame

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateInformationDialog

#### FORMAT

widget XmCreateInformationDialog  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateLabel

#### FORMAT

widget XmCreateLabel  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |



| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

## XmCreateLabelGadget

### FORMAT

widget XmCreateLabelGadget  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateList

### FORMAT

widget XmCreateList  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

## XmCreateMainWindow

### FORMAT

widget XmCreateMainWindow  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateMenuBar

### FORMAT

widget XmCreateMenuBar  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateMenuShell

### FORMAT

widget XmCreateMenuShell  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**XmCreateMessageBox**

**FORMAT**

widget XmCreateMessageBox  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**XmCreateMessageDialog**

**FORMAT**

widget XmCreateMessageDialog  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

## OSF/Motif Toolkit Routines

### XmCreateMessageDialog

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

---

## XmCreateOptionMenu

### FORMAT

widget XmCreateOptionMenu  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreatePanedWindow

### FORMAT

widget XmCreatePanedWindow  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreatePopupMenu

### FORMAT

widget XmCreatePopupMenu  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreatePromptDialog

### FORMAT

widget XmCreatePromptDialog  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreatePulldownMenu

### FORMAT

widget XmCreatePulldownMenu  
(parent, name, arglist, argcount)

## OSF/Motif Toolkit Routines

### XmCreatePulldownMenu

#### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreatePushButton

#### FORMAT

widget XmCreatePushButton  
(parent, name, arglist, argcount)

#### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreatePushButtonGadget

#### FORMAT

widget XmCreatePushButtonGadget  
(parent, name, arglist, argcount)

#### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

---

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

---

## XmCreateQuestionDialog

### FORMAT

widget XmCreateQuestionDialog  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateRadioBox

### FORMAT

widget XmCreateRadioBox  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateRowColumn

### FORMAT

widget XmCreateRowColumn  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateScale

### FORMAT

widget XmCreateScale  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateScrollBar

### FORMAT

widget XmCreateScrollBar  
(parent, name, arglist, argcount)



**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**XmCreateScrolledList**

**FORMAT**

widget XmCreateScrolledList  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**XmCreateScrolledText**

**FORMAT**

widget XmCreateScrolledText  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

## OSF/Motif Toolkit Routines

### XmCreateScrolledText

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

---

## XmCreateScrolledWindow

### FORMAT

widget XmCreateScrolledWindow  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateSelectionBox

### FORMAT

widget XmCreateSelectionBox  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateSelectionDialog

### FORMAT

widget XmCreateSelectionDialog  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateSeparator

### FORMAT

widget XmCreateSeparator  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateSeparatorGadget

### FORMAT

widget XmCreateSeparatorGadget  
(parent, name, arglist, argcount)

## OSF/Motif Toolkit Routines

### XmCreateSeparatorGadget

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateSimpleCheckBox

#### FORMAT

result XmCreateSimpleCheckBox  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateSimpleMenuBar

#### FORMAT

result XmCreateSimpleMenuBar  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

## XmCreateSimpleOptionMenu

### FORMAT

result XmCreateSimpleOptionMenu  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateSimplePopupMenu

### FORMAT

result XmCreateSimplePopupMenu  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

## XmCreateSimplePulldownMenu

### FORMAT

result XmCreateSimplePulldownMenu  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreatePushButtonGadget

### FORMAT

widget XmCreatePushButtonGadget  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateSimpleRadioBox

### FORMAT

result XmCreateSimpleRadioBox  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateTemplateDialog

**FORMAT**

widget XmCreateTemplateDialog  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateText

**FORMAT**

widget XmCreateText  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

## OSF/Motif Toolkit Routines

### XmCreateText

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

---

## XmCreateTextField

### FORMAT

widget XmCreateTextField  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

---

## XmCreateToggleButton

### FORMAT

widget XmCreateToggleButton  
(parent, name, arglist, argcount)

### Argument Information

---

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---



---

## XmCreateToggleButtonGadget

### FORMAT

widget XmCreateToggleButtonGadget  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateWarningDialog

### FORMAT

widget XmCreateWarningDialog  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateWorkArea

### FORMAT

widget XmCreateWorkArea  
(parent, name, arglist, argcount)

## OSF/Motif Toolkit Routines

### XmCreateWorkArea

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateWorkingDialog

#### FORMAT

widget XmCreateWorkingDialog  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

## XmCreateWorkArea

#### FORMAT

widget XmCreateWorkArea  
(parent, name, arglist, argcount)

#### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| argcount | Cardinal | read   | value     |

---

## XmCreateWorkingDialog

### FORMAT

widget XmCreateWorkingDialog  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | Arg      | read   | reference |
| argcount | Cardinal | read   | value     |

---

## XmCvtCTToXmString

### FORMAT

result XmCvtCTToXmString  
(text)

### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| result   | XmString  | write  | value     |
| text     | CHARACTER | read   | reference |

---

## XmCvtStringToUnitType

### FORMAT

XmCvtStringToUnitType  
(args, num\_args, from\_val, to\_val)

### Argument Information

| Argument | Type                | Access | Mechanism |
|----------|---------------------|--------|-----------|
| args     | XrmValuePtr (array) | read   | reference |
| num_args | Cardinal            | read   | reference |
| from_val | XrmValuePtr         | read   | reference |
| to_val   | XrmValuePtr         | read   | reference |

---

## XmCvtXmStringToCT

### FORMAT

result XmCvtXmStringToCT  
(string)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| string   | XmString | read   | value     |

---

## XmDeactivateProtocol

### FORMAT

XmDeactivateProtocol  
(shell, property, protocol )

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| shell    | Widget | read   | value     |

| Argument | Type | Access | Mechanism |
|----------|------|--------|-----------|
| property | Atom | read   | value     |
| protocol | Atom | read   | value     |

---

## XmDestroyPixmap

### FORMAT

result XmDestroyPixmap  
(screen, pixmap)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| screen   | Screen  | read   | reference |
| pixmap   | Pixmap  | read   | value     |

---

## XmDragCancel

### FORMAT

XmDragCancel  
(dragContext)

### Argument Information

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| dragContext | Widget | read   | value     |

---

## XmDragStart

### FORMAT

result XmDragStart  
(widget, display, args, numArgs)

## OSF/Motif Toolkit Routines

### XmDragStart

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | Widget          | write  | value     |
| widget   | Widget          | read   | value     |
| display  | ANY             | read   | reference |
| args     | ArgList (array) | read   | reference |
| numArgs  | Cardinal        | read   | value     |

---

## XmDropSiteConfigureStackingOrde

#### FORMAT

XmDropSiteConfigureStackingOrde  
(widget, sibling, stack\_mode)

#### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| widget     | Widget   | read   | value     |
| sibling    | Widget   | read   | value     |
| stack_mode | Cardinal | read   | value     |

---

## XmDropSiteEndUpdate

#### FORMAT

XmDropSiteEndUpdate  
(refwidget)

#### Argument Information

| Argument  | Type   | Access | Mechanism |
|-----------|--------|--------|-----------|
| refwidget | Widget | read   | value     |

---

## XmDropSiteGetActiveVisuals

### FORMAT

result XmDropSiteGetActiveVisuals  
(widget)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | XmDropSiteVisuals | write  | value     |
| widget   | Widget            | read   | value     |

---

## XmDropSiteQueryStackingOrder

### FORMAT

result XmDropSiteQueryStackingOrder  
(widget, parent\_rtn, children\_rtn, num\_children\_rtn)

### Argument Information

| Argument         | Type     | Access | Mechanism |
|------------------|----------|--------|-----------|
| result           | LONGWORD | write  | value     |
| widget           | Widget   | read   | value     |
| parent_rtn       | Widget   | write  | reference |
| children_rtn     | ADDRESS  | write  | reference |
| num_children_rtn | Cardinal | write  | reference |

---

## XmDropSiteRegister

### FORMAT

XmDropSiteRegister  
(widget, args, argCount)

## OSF/Motif Toolkit Routines

### XmDropSiteRegister

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | Widget          | read   | value     |
| args     | ArgList (array) | read   | reference |
| argCount | Cardinal        | read   | value     |

---

## XmDropSiteRetrieve

### FORMAT

XmDropSiteRetrieve  
(enclosingwidget, args, argCount)

#### Argument Information

| Argument        | Type            | Access | Mechanism |
|-----------------|-----------------|--------|-----------|
| enclosingwidget | Widget          | read   | value     |
| args            | ArgList (array) | read   | reference |
| argCount        | Cardinal        | read   | value     |

---

## XmDropSiteStartUpdate

### FORMAT

XmDropSiteStartUpdate  
(refwidget)

#### Argument Information

| Argument  | Type   | Access | Mechanism |
|-----------|--------|--------|-----------|
| refwidget | Widget | read   | value     |



---

## XmDropSiteUnregister

### FORMAT

XmDropSiteUnregister  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

## XmDropSiteUpdate

### FORMAT

XmDropSiteUpdate  
(enclosingwidget, args, argCount)

### Argument Information

| Argument        | Type            | Access | Mechanism |
|-----------------|-----------------|--------|-----------|
| enclosingwidget | Widget          | read   | value     |
| args            | ArgList (array) | read   | reference |
| argCount        | Cardinal        | read   | value     |

---

## XmDropTransferAdd

### FORMAT

XmDropTransferAdd  
(widget, transfers, num\_transfers)

### Argument Information

| Argument      | Type                | Access | Mechanism |
|---------------|---------------------|--------|-----------|
| widget        | Widget              | read   | value     |
| transfers     | XmDropTransferEntry | read   | value     |
| num_transfers | Cardinal            | read   | value     |

## XmDropTransferStart

### FORMAT

widget XmDropTransferStart  
(refWidget, args, argCount)

### Argument Information

| Argument  | Type        | Access | Mechanism |
|-----------|-------------|--------|-----------|
| widget    | Widget      | write  | value     |
| refWidget | Widget      | read   | value     |
| args      | Arg (array) | read   | reference |
| argCount  | Cardinal    | read   | value     |

---

## XmFileSelectionBoxGetChild

### FORMAT

result XmFileSelectionBoxGetChild  
(widget, child)

### Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | Widget        | write  | value     |
| widget   | Widget        | read   | value     |
| child    | BYTE UNSIGNED | read   | value     |

---

## XmFileSelectionDoSearch

### FORMAT

XmFileSelectionDoSearch  
(widget, dirmask)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| dirmask  | XmString | read   | value     |

---

**XmFontListAdd**

**FORMAT**

result XmFontListAdd  
(oldlist, font, charset)

**Argument Information**

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| result   | XmFontList | write  | value     |
| oldlist  | XmFontList | read   | value     |
| font     | Font       | read   | value     |
| charset  | XtString   | read   | reference |

---

**XmFontListAppendEntry**

**FORMAT**

result XmFontListAppendEntry  
(old, entry)

**Argument Information**

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmFontList      | write  | value     |
| old      | XmFontList      | read   | value     |
| entry    | XmFontListEntry | read   | value     |

## XmFontListCopy

### FORMAT

result XmFontListCopy  
(fontlist)

### Argument Information

---

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| result   | XmFontList | write  | value     |
| fontlist | XmFontList | read   | value     |

---

---

## XmFontListCreate

### FORMAT

result XmFontListCreate  
(font, charset)

### Argument Information

---

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| result   | XmFontList | write  | value     |
| font     | Font       | read   | value     |
| charset  | XtString   | read   | reference |

---

---

## XmFontListEntryCreate

### FORMAT

result XmFontListEntryCreate  
(tag, type, font)

### Argument Information

---

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmFontListEntry | write  | value     |
| tag      | XtString        | read   | reference |

---

---

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| type     | XmFontType | read   | value     |
| font     | XtPointer  | read   | value     |

---

---

## XmFontListEntryFree

### FORMAT

XmFontListEntryFree  
(entry)

### Argument Information

---

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| entry    | XmFontListEntry | write  | reference |

---

---

## XmFontListEntryGetFont

### FORMAT

result XmFontListEntryGetFont  
(entry, typeReturn)

### Argument Information

---

| Argument   | Type            | Access | Mechanism |
|------------|-----------------|--------|-----------|
| result     | XtPointer       | write  | value     |
| entry      | XmFontListEntry | read   | value     |
| typeReturn | XmFontType      | write  | reference |

---

---

## XmFontListEntryGetTag

### FORMAT

result XmFontListEntryGetTag  
(entry)

## OSF/Motif Toolkit Routines

### XmFontListEntryGetTag

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XtString        | write  | value     |
| entry    | XmFontListEntry | read   | value     |

---

## XmFontListEntryLoad

### FORMAT

result XmFontListEntryLoad  
(display, fontName, type, tag)

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmFontListEntry | write  | value     |
| display  | Display         | read   | value     |
| fontName | XtString        | read   | reference |
| type     | XmFontType      | read   | value     |
| tag      | XtString        | read   | reference |

---

## XmFontListFree

### FORMAT

XmFontListFree  
(fontlist)

#### Argument Information

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| fontlist | XmFontList | read   | value     |

---

## XmFontListFreeFontContext

### FORMAT

XmFontListFreeFontContext  
(context)

### Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| context  | XmFontContext | read   | value     |

---

## XmFontListGetNextFont

### FORMAT

result XmFontListGetNextFont  
(context, charset, font)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | Boolean         | write  | value     |
| context  | XmFontContext   | read   | value     |
| charset  | XmStringCharSet | write  | reference |
| font     | Font            | read   | reference |

---

## XmFontListInitFontContext

### FORMAT

result XmFontListInitFontContext  
(context, fontlist)

### Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | Boolean       | write  | value     |
| context  | XmFontContext | read   | reference |

## OSF/Motif Toolkit Routines

### XmFontListInitFontContext

---

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| fontlist | XmFontList | read   | value     |

---

---

### XmFontListNextEntry

#### FORMAT

result XmFontListNextEntry  
(context)

#### Argument Information

---

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmFontListEntry | write  | value     |
| context  | XmFontContext   | read   | value     |

---

---

### XmFontListRemoveEntry

#### FORMAT

result XmFontListRemoveEntry  
(old, entry)

#### Argument Information

---

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmFontList      | write  | value     |
| old      | XmFontList      | read   | value     |
| entry    | XmFontListEntry | read   | value     |

---

---

### XmGetAtomName

#### FORMAT

result XmGetAtomName  
(display, atom)



**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| display  | Display  | read   | value     |
| atom     | Atom     | read   | value     |

---

**XmGetColorCalculation**

**FORMAT**

result XmGetColorCalculation  
( )

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | XmColorProc | write  | value     |

---

**XmGetColors**

**FORMAT**

XmGetColors  
(screen, colormap, background, foreground, top\_shadow, bottom\_shadow, select)

**Argument Information**

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| screen        | Screen   | read   | reference |
| colormap      | Colormap | read   | value     |
| background    | Pixel    | read   | value     |
| foreground    | Pixel    | write  | reference |
| top_shadow    | Pixel    | write  | reference |
| bottom_shadow | Pixel    | write  | reference |
| select        | Pixel    | write  | reference |

---

## XmGetDestination

### FORMAT

result XmGetDestination  
(display)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Widget  | write  | value     |
| display  | Display | read   | value     |

---

## XmGetDragContext

### FORMAT

result XmGetDragContext  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

---

## XmGetFocusWidget

### FORMAT

widget XmGetFocusWidget  
(wid)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | write  | value     |
| wid      | Widget | read   | value     |

---

## XmGetMenuCursor

### FORMAT

result XmGetMenuCursor  
(display)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Cursor  | write  | value     |
| display  | Display | read   | value     |

---

## XmGetPixmap

### FORMAT

result XmGetPixmap  
(screen, image\_name, foreground, background)

### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | Pixmap   | write  | value     |
| screen     | Screen   | read   | reference |
| image_name | XtString | read   | reference |
| foreground | Pixel    | read   | value     |
| background | Pixel    | read   | value     |

---

## XmGetPixmapByDepth

### FORMAT

result XmGetPixmapByDepth  
(screen, image\_name, foreground, background, depth)

## OSF/Motif Toolkit Routines

### XmGetPixmapByDepth

#### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | Pixmap   | write  | value     |
| screen     | Screen   | read   | reference |
| image_name | XtString | read   | reference |
| foreground | Pixel    | read   | value     |
| background | Pixel    | read   | value     |
| depth      | LONGWORD | read   | value     |

---

## XmGetPostedFromWidget

### FORMAT

widget XmGetPostedFromWidget  
(menu)

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | write  | value     |
| menu     | Widget | read   | value     |

---

## XmGetTabGroup

### FORMAT

widget XmGetTabGroup  
(wid)

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | write  | value     |
| wid      | Widget | read   | value     |

---

## XmGetTearOffControl

### FORMAT

widget XmGetTearOffControl  
(menu)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | write  | value     |
| menu     | Widget | read   | value     |

---

## XmGetVisibility

### FORMAT

result XmGetVisibility  
(wid)

### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| result   | XmVisibility | write  | value     |
| wid      | Widget       | read   | value     |

---

## XmGetXmDisplay

### FORMAT

widget XmGetXmDisplay  
(display)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | write  | value     |
| display  | Display | write  | reference |

## XmGetXmScreen

### FORMAT

widget XmGetXmScreen  
(screen)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | write  | value     |
| screen   | Screen | read   | reference |

---

## XmImGetXIM

### FORMAT

result XmImGetXIM  
(w)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | ADDRESS | write  | value     |
| w        | Widget  | read   | value     |

---

## XmImMbLookupString

### FORMAT

result XmImMbLookupString  
(w, event, buf, nbytes, keysym, status)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| w        | Widget   | read   | write     |
| event    | ANY      | read   | reference |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| buf      | XtString | read   | reference |
| nbytes   | LONGWORD | read   | value     |
| keysym   | KeySym   | write  | reference |
| status   | LONGWORD | write  | reference |

---

## XmImRegister

### FORMAT

XmImRegister  
(w, reserved)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| w        | Widget   | read   | value     |
| reserved | LONGWORD | read   | value     |

---

## XmImSetFocusValues

### FORMAT

XmImSetFocusValues  
(w, args, num\_args)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| w        | Widget      | read   | value     |
| args     | Arg (array) | read   | reference |
| num_args | Cardinal    | read   | value     |

---

## XmImSetValues

### FORMAT

XmImSetValues  
(w, args, num\_args)

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| w        | Widget      | read   | value     |
| args     | Arg (array) | read   | reference |
| num_args | Cardinal    | read   | value     |

---

## XmImUnregister

### FORMAT

XmImUnregister  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XmImUnsetFocus

### FORMAT

XmImUnsetFocus  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |



---

## XmlmVaSetFocusValues

### FORMAT

XmlmVaSetFocusValues  
 (w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XmlmVaSetValues

### FORMAT

XmlmVaSetValues  
 (w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XmlInstallImage

### FORMAT

result XmlInstallImage  
 (image, image\_name)

### Argument Information

| Argument   | Type      | Access | Mechanism |
|------------|-----------|--------|-----------|
| result     | Boolean   | write  | value     |
| image      | XImage    | read   | reference |
| image_name | CHARACTER | read   | reference |

## OSF/Motif Toolkit Routines

### XmlInternAtom

---

## XmlInternAtom

### FORMAT

result XmlInternAtom  
(display, name, only\_if\_exists)

### Argument Information

| Argument       | Type                 | Access | Mechanism |
|----------------|----------------------|--------|-----------|
| result         | LONGWORD<br>UNSIGNED | write  | value     |
| display        | Display              | read   | value     |
| name           | XtString             | read   | reference |
| only_if_exists | Boolean              | read   | value     |

---

## XmlsMotifWMRunning

### FORMAT

result XmlsMotifWMRunning  
(shell)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| shell    | Widget  | read   | value     |

---

## XmlsTraversable

### FORMAT

result XmlsTraversable  
(wid)

**Argument Information**

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| wid      | Widget  | read   | value     |

---

**XmListAddItem**

**FORMAT**

XmListAddItem  
(widget, item, position)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |
| position | LONGWORD | read   | value     |

---

**XmListAddItems**

**FORMAT**

XmListAddItems  
(widget, items, item\_count, pos)

**Argument Information**

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| widget     | Widget           | read   | value     |
| items      | XmString (array) | read   | reference |
| item_count | LONGWORD         | read   | value     |
| pos        | LONGWORD         | read   | value     |

---

## XmListAddItemsUnselected

### FORMAT

XmListAddItemsUnselected  
(widget, items, item\_count, pos)

### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| widget     | Widget           | read   | value     |
| items      | XmString (array) | read   | reference |
| item_count | LONGWORD         | read   | value     |
| pos        | LONGWORD         | read   | value     |

---

## XmListAddItemUnselected

### FORMAT

XmListAddItemUnselected  
(widget, item, pos)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |
| pos      | LONGWORD | read   | value     |

---

## XmListDeleteAllItems

### FORMAT

XmListDeleteAllItems  
(widget)

**Argument Information**

---

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| widget          | Widget      | read          | value            |

---

---

**XmListDeleteItem**

**FORMAT**

XmListDeleteItem  
(widget, item)

**Argument Information**

---

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| widget          | Widget      | read          | value            |
| item            | XmString    | read          | value            |

---

---

**XmListDeleteItems**

**FORMAT**

XmListDeleteItems  
(widget, items, item\_count)

**Argument Information**

---

| <b>Argument</b> | <b>Type</b>      | <b>Access</b> | <b>Mechanism</b> |
|-----------------|------------------|---------------|------------------|
| widget          | Widget           | read          | value            |
| items           | XmString (array) | read          | reference        |
| item_count      | LONGWORD         | read          | value            |

---

---

## XmListDeleteItemsPos

### FORMAT

XmListDeleteItemsPos  
(widget, item\_count, pos)

### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| widget     | Widget   | read   | value     |
| item_count | LONGWORD | read   | value     |
| pos        | LONGWORD | read   | value     |

---

## XmListDeletePos

### FORMAT

XmListDeletePos  
(widget, pos)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| pos      | LONGWORD | read   | value     |

---

## XmListDeletePositions

### FORMAT

XmListDeletePositions  
(w, position\_list, position\_count)

### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| w             | Widget   | read   | value     |
| position_list | LONGWORD | write  | reference |

---

| Argument       | Type     | Access | Mechanism |
|----------------|----------|--------|-----------|
| position_count | LONGWORD | read   | value     |

---

---

## XmListDeselectAllItems

### FORMAT

XmListDeselectAllItems  
(widget)

### Argument Information

---

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

---

## XmListDeselectItem

### FORMAT

XmListDeselectItem  
(widget, item)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |

---

---

## XmListDeselectPos

### FORMAT

XmListDeselectPos  
(widget, pos)

## OSF/Motif Toolkit Routines

### XmListDeselectPos

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| pos      | LONGWORD | read   | value     |

---

## XmListGetKbdtemPos

### FORMAT

result XmListGetKbdtemPos  
(widget)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmListGetMatchPos

### FORMAT

XmListGetMatchPos  
(widget, item, position\_list, position\_count)

#### Argument Information

| Argument       | Type                  | Access | Mechanism |
|----------------|-----------------------|--------|-----------|
| widget         | Widget                | read   | value     |
| item           | XmString              | read   | value     |
| position_list  | ADDRESS<br>(LONGWORD) | write  | reference |
| position_count | LONGWORD              | write  | reference |



---

## XmListGetSelectedPos

### FORMAT

result XmListGetSelectedPos  
(widget, pos\_list, pos\_count)

### Argument Information

| Argument  | Type                  | Access | Mechanism |
|-----------|-----------------------|--------|-----------|
| result    | Boolean               | write  | value     |
| widget    | Widget                | read   | value     |
| pos_list  | ADDRESS<br>(LONGWORD) | write  | reference |
| pos_count | LONGWORD              | write  | reference |

---

## XmListItemExists

### FORMAT

result XmListItemExists  
(widget, item)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |

---

## XmListItemPos

### FORMAT

result XmListItemPos  
(widget, item)

## OSF/Motif Toolkit Routines

### XmListItemPos

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |

---

## XmListPosSelected

### FORMAT

result XmListPosSelected  
(widget, position)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| widget   | Widget   | read   | value     |
| position | LONGWORD | read   | value     |

---

## XmListPosToBounds

### FORMAT

result XmListPosToBounds  
(widget, position, x, y, width, height)

#### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| result   | Boolean   | write  | value     |
| widget   | Widget    | read   | value     |
| position | LONGWORD  | read   | value     |
| x        | Position  | write  | reference |
| y        | Position  | write  | reference |
| width    | Dimension | write  | reference |
| height   | Dimension | write  | reference |

---

## XmListReplaceltems

### FORMAT

XmListReplaceltems  
(widget, new\_items, item\_count, position)

### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| widget     | Widget           | read   | value     |
| new_items  | XmString (array) | read   | reference |
| item_count | LONGWORD         | read   | value     |
| position   | LONGWORD         | read   | value     |

---

## XmListReplaceltemsPos

### FORMAT

XmListReplaceltemsPos  
(widget, new\_items, item\_count, position)

### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| widget     | Widget           | read   | value     |
| new_items  | XmString (array) | read   | reference |
| item_count | LONGWORD         | read   | value     |
| position   | LONGWORD         | read   | value     |

---

## XmListReplaceltemsPosUnselected

### FORMAT

XmListReplaceltemsPosUnselected  
(widget, new\_items, item\_count, position)

## OSF/Motif Toolkit Routines

### XmListReplaceltemsPosUnselected

#### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| widget     | Widget           | read   | value     |
| new_items  | XmString (array) | read   | reference |
| item_count | LONGWORD         | read   | value     |
| position   | LONGWORD         | read   | value     |

---

## XmListReplaceltemsUnselected

### FORMAT

XmListReplaceltemsUnselected  
(w, old\_items, item\_count, new\_items)

#### Argument Information

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| w          | Widget           | read   | value     |
| old_items  | XmString (array) | read   | reference |
| item_count | LONGWORD         | read   | value     |
| new_items  | XmString (array) | read   | reference |

---

## XmListReplacePositions

### FORMAT

XmListReplacePositions  
(w, position\_list, item\_list, item\_count)

#### Argument Information

| Argument      | Type             | Access | Mechanism |
|---------------|------------------|--------|-----------|
| w             | Widget           | read   | value     |
| position_list | LONGWORD         | write  | reference |
| item_list     | XmString (array) | read   | reference |
| item_count    | LONGWORD         | read   | value     |

---

## XmListSelectItem

### FORMAT

XmListSelectItem  
(widget, item, notify)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |
| notify   | Boolean  | read   | value     |

---

## XmListSelectPos

### FORMAT

XmListSelectPos  
(widget, pos, notify)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| pos      | LONGWORD | read   | value     |
| notify   | Boolean  | read   | value     |

---

## XmListSetAddMode

### FORMAT

XmListSetAddMode  
(widget, mode)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

## OSF/Motif Toolkit Routines

### XmListSetAddMode

---

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| mode     | Boolean | read   | value     |

---

---

### XmListSetBottomItem

#### FORMAT

XmListSetBottomItem  
(widget, item)

#### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |

---

---

### XmListSetBottomPos

#### FORMAT

XmListSetBottomPos  
(widget, position)

#### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| position | LONGWORD | read   | value     |

---

---

### XmListSetHorizPos

#### FORMAT

XmListSetHorizPos  
(widget, position)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| position | LONGWORD | read   | value     |

---

**XmListSetItem**

**FORMAT**

XmListSetItem  
(widget, item)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| item     | XmString | read   | value     |

---

**XmListSetKbdItemPos**

**FORMAT**

result XmListSetKbdItemPos  
(widget, position)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| widget   | Widget   | read   | value     |
| position | LONGWORD | read   | value     |

## XmListSetPos

### FORMAT

XmListSetPos  
(widget, pos)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| pos      | LONGWORD | read   | value     |

---

## XmListUpdateSelectedList

### FORMAT

XmListUpdateSelectedList  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

## XmListYToPos

### FORMAT

result XmListYToPos  
(widget, y)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |
| y        | Position | read   | value     |

---



---

## XmMainWindowSep1

### FORMAT

result XmMainWindowSep1  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |
| widget   | Widget | read   | value     |

---

## XmMainWindowSep2

### FORMAT

result XmMainWindowSep2  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |
| widget   | Widget | read   | value     |

---

## XmMainWindowSep3

### FORMAT

result XmMainWindowSep3  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |
| widget   | Widget | read   | value     |

---

## XmMainWindowSetAreas

### FORMAT

XmMainWindowSetAreas

(widget, menu\_bar, command\_window, horizontal\_scrollbar, vertical\_scrollbar,  
work\_region)

### Argument Information

| Argument             | Type   | Access | Mechanism |
|----------------------|--------|--------|-----------|
| widget               | Widget | read   | value     |
| menu_bar             | Widget | read   | value     |
| command_window       | Widget | read   | value     |
| horizontal_scrollbar | Widget | read   | value     |
| vertical_scrollbar   | Widget | read   | value     |
| work_region          | Widget | read   | value     |

---

## XmMapSegmentEncoding

### FORMAT

result XmMapSegmentEncoding

(fontlist\_tag)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| result       | XtString | write  | value     |
| fontlist_tag | XtString | read   | reference |

---

## XmMenuPosition

### FORMAT

XmMenuPosition

(p, event)

Argument Information

| Argument | Type                | Access | Mechanism |
|----------|---------------------|--------|-----------|
| p        | Widget              | read   | value     |
| event    | XButtonPressedEvent | read   | reference |

---

## XmMessageBoxGetChild

### FORMAT

result XmMessageBoxGetChild  
(widget, child)

Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | Widget        | write  | value     |
| widget   | Widget        | read   | value     |
| child    | BYTE UNSIGNED | read   | value     |

---

## XmOptionButtonGadget

### FORMAT

widget XmOptionButtonGadget  
(option\_menu)

Argument Information

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| widget      | Widget | write  | value     |
| option_menu | Widget | read   | value     |

---

## XmOptionLabelGadget

### FORMAT

widget XmOptionLabelGadget  
(option\_menu)

### Argument Information

| Argument    | Type   | Access | Mechanism |
|-------------|--------|--------|-----------|
| widget      | Widget | write  | value     |
| option_menu | Widget | read   | value     |

---

## XmProcessTraversal

### FORMAT

result XmProcessTraversal  
(widget, direction)

### Argument Information

| Argument  | Type                 | Access | Mechanism |
|-----------|----------------------|--------|-----------|
| result    | Boolean              | write  | value     |
| widget    | Widget               | read   | value     |
| direction | XmTraversalDirection | read   | value     |

---

## XmRegisterConverters

### FORMAT

XmRegisterConverters

---

## XmRegisterSegmentEncoding

### FORMAT

result XmRegisterSegmentEncoding  
(fontlist\_tag, ct\_encoding)

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| result          | XtString    | write         | value            |
| fontlist_tag    | XtString    | read          | reference        |
| ct_encoding     | XtString    | read          | reference        |

## **XmRemoveFromPostFromList**

**FORMAT**

XmRemoveFromPostFromList  
(menu\_wid, widget)

**Argument Information**

| <b>Argument</b> | <b>Type</b> | <b>Access</b> | <b>Mechanism</b> |
|-----------------|-------------|---------------|------------------|
| menu_wid        | Widget      | read          | value            |
| widget          | Widget      | read          | value            |

## **XmRemoveProtocolCallback**

**FORMAT**

XmRemoveProtocolCallback  
(shell, property, protocol, callback, closure)

**Argument Information**

| <b>Argument</b> | <b>Type</b>    | <b>Access</b> | <b>Mechanism</b> |
|-----------------|----------------|---------------|------------------|
| shell           | Widget         | read          | value            |
| property        | Atom           | read          | value            |
| protocol        | Atom           | read          | value            |
| callback        | XtCallbackProc | read          | value            |
| closure         | XtPointer      | read          | value            |

---

## XmRemoveProtocols

### FORMAT

XmRemoveProtocols  
(shell, property, protocols, num\_protocols)

### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| shell         | Widget   | read   | value     |
| property      | Atom     | read   | value     |
| protocols     | Atom     | read   | reference |
| num_protocols | Cardinal | read   | value     |

---

## XmRemoveTabGroup

### FORMAT

XmRemoveTabGroup  
(w)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| w        | Widget | read   | value     |

---

## XmRepTypeAddReverse

### FORMAT

XmRepTypeAddReverse  
(rep\_type\_id)

### Argument Information

| Argument    | Type        | Access | Mechanism |
|-------------|-------------|--------|-----------|
| rep_type_id | XmRepTypeId | read   | value     |

---

## XmRepTypeGetId

### FORMAT

```
result XmRepTypeGetId
    (rep_type)
```

### Argument Information

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | XmRepTypeId | write  | value     |
| rep_type | XtString    | read   | reference |

---

## XmRepTypeGetNameList

### FORMAT

```
result XmRepTypeGetNameList
    (rep_type_id, use_uppercase_format)
```

### Argument Information

| Argument             | Type        | Access | Mechanism |
|----------------------|-------------|--------|-----------|
| result               | ADDRESS     | write  | value     |
| rep_type_id          | XmRepTypeId | read   | value     |
| use_uppercase_format | Boolean     | read   | value     |

---

## XmRepTypeGetRecord

### FORMAT

```
result XmRepTypeGetRecord
    (rep_type_id)
```

### Argument Information

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| result      | XmRepTypeEntry | write  | value     |
| rep_type_id | XmRepTypeId    | read   | value     |

---

## XmRepTypeGetRegistered

### FORMAT

result XmRepTypeGetRegistered

### Argument Information

---

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | XmRepTypeList | write  | value     |

---

---

## XmRepTypeInstallTearOffModelConverter

### FORMAT

XmRepTypeInstallTearOffModelConverter

---

## XmRepTypeRegister

### FORMAT

result XmRepTypeRegister  
(rep\_type, value\_names, values, num\_values)

### Argument Information

---

| Argument    | Type          | Access | Mechanism |
|-------------|---------------|--------|-----------|
| result      | XmRepTypeId   | write  | value     |
| rep_type    | XtString      | read   | reference |
| value_names | ADDRESS       | write  | reference |
| values      | XtString      | read   | reference |
| num_values  | BYTE UNSIGNED | read   | value     |

---

---

## XmRepTypeValidValue

### FORMAT

result XmRepTypeValidValue  
(rep\_type\_id, test\_value, enable\_default\_warning)



**Argument Information**

| Argument               | Type          | Access | Mechanism |
|------------------------|---------------|--------|-----------|
| result                 | Boolean       | write  | value     |
| rep_type_id            | XmRepTypeId   | read   | value     |
| test_value             | BYTE UNSIGNED | read   | value     |
| enable_default_warning | Widget        | read   | value     |

---

**XmResolveAllPartOffsets**

**FORMAT**

XmResolveAllPartOffsets  
(wclass, offset, constraint\_offset)

**Argument Information**

| Argument          | Type        | Access | Mechanism |
|-------------------|-------------|--------|-----------|
| wclass            | WidgetClass | read   | value     |
| offset            | XmOffsetPtr | modify | reference |
| constraint_offset | XmOffsetPtr | modify | reference |

---

**XmResolvePartOffsets**

**FORMAT**

XmResolvePartOffsets  
(widget\_class, offset)

**Argument Information**

| Argument     | Type        | Access | Mechanism |
|--------------|-------------|--------|-----------|
| widget_class | WidgetClass | read   | value     |
| offset       | XmOffsetPtr | modify | reference |

---

## XmScaleGetValue

### FORMAT

XmScaleGetValue  
(widget, value)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| value    | LONGWORD | write  | reference |

---

## XmScaleSetValue

### FORMAT

XmScaleSetValue  
(widget, value)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| value    | LONGWORD | read   | value     |

---

## XmScrollBarGetValues

### FORMAT

XmScrollBarGetValues  
(widget, value\_return, slider\_size\_return, increment\_return, page\_increment\_return)

### Argument Information

| Argument           | Type     | Access | Mechanism |
|--------------------|----------|--------|-----------|
| widget             | Widget   | read   | value     |
| value_return       | LONGWORD | write  | reference |
| slider_size_return | LONGWORD | write  | reference |

| Argument              | Type     | Access | Mechanism |
|-----------------------|----------|--------|-----------|
| increment_return      | LONGWORD | write  | reference |
| page_increment_return | LONGWORD | write  | reference |

---

## XmScrollBarSetValues

### FORMAT

XmScrollBarSetValues

(widget, value, slider\_size, increment, page\_increment, notify)

### Argument Information

| Argument       | Type     | Access | Mechanism |
|----------------|----------|--------|-----------|
| widget         | Widget   | read   | value     |
| value          | LONGWORD | read   | value     |
| slider_size    | LONGWORD | read   | value     |
| increment      | LONGWORD | read   | value     |
| page_increment | LONGWORD | read   | value     |
| notify         | Boolean  | read   | value     |

---

## XmScrollVisible

### FORMAT

XmScrollVisible

(widget, widget\_id, hor\_margin, ver\_margin)

### Argument Information

| Argument   | Type      | Access | Mechanism |
|------------|-----------|--------|-----------|
| widget     | Widget    | read   | value     |
| widget_id  | Widget    | read   | value     |
| hor_margin | Dimension | read   | value     |
| ver_margin | Dimension | read   | value     |

## XmScrolledWindowSetAreas

### FORMAT

XmScrolledWindowSetAreas  
(widget, horizontal\_scrollbar, vertical\_scrollbar, work\_region)

### Argument Information

| Argument             | Type   | Access | Mechanism |
|----------------------|--------|--------|-----------|
| widget               | Widget | read   | value     |
| horizontal_scrollbar | Widget | read   | value     |
| vertical_scrollbar   | Widget | read   | value     |
| work_region          | Widget | read   | value     |

---

## XmSelectionBoxGetChild

### FORMAT

result XmSelectionBoxGetChild  
(widget, child)

### Argument Information

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| result   | Widget        | write  | value     |
| widget   | Widget        | read   | value     |
| child    | BYTE UNSIGNED | read   | value     |

---

## XmSetColorCalculation

### FORMAT

result XmSetColorCalculation  
(proc)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| result   | XmColorProc | write  | value     |
| proc     | XmColorProc | read   | value     |

---

**XmSetFontUnit**

**FORMAT**

XmSetFontUnit  
(display, font\_unit\_value)

**Argument Information**

| Argument        | Type     | Access | Mechanism |
|-----------------|----------|--------|-----------|
| display         | Display  | read   | value     |
| font_unit_value | LONGWORD | read   | value     |

---

**XmSetFontUnits**

**FORMAT**

XmSetFontUnits  
(display, h\_value, v\_value)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| display  | Display  | read   | value     |
| h_value  | LONGWORD | read   | value     |
| v_value  | LONGWORD | read   | value     |

## OSF/Motif Toolkit Routines

### XmSetMenuCursor

---

## XmSetMenuCursor

### FORMAT

XmSetMenuCursor  
(display, cursorId)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| display  | Display | read   | value     |
| cursorId | Cursor  | read   | value     |

---

## XmSetProtocolHooks

### FORMAT

XmSetProtocolHooks  
(shell, property, protocol, prehook, pre\_closure, posthook, post\_closure)

### Argument Information

| Argument     | Type           | Access | Mechanism |
|--------------|----------------|--------|-----------|
| shell        | Widget         | read   | value     |
| property     | Atom           | read   | value     |
| protocol     | Atom           | read   | value     |
| prehook      | XtCallbackProc | read   | value     |
| pre_closure  | XtPointer      | read   | value     |
| posthook     | XtCallbackProc | read   | value     |
| post_closure | XtPointer      | read   | value     |

---

## XmStringBaseline

### FORMAT

result XmStringBaseline  
(fontlist, string)

**Argument Information**

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| result   | Dimension  | write  | value     |
| fontlist | XmFontList | read   | value     |
| string   | XmString   | read   | value     |

---

**XmStringByteCompare**

**FORMAT**

result XmStringByteCompare  
(s1, s2)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| s1       | XmString | read   | value     |
| s2       | XmString | read   | value     |

---

**XmStringCompare**

**FORMAT**

result XmStringCompare  
(s1, s2)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| s1       | XmString | read   | value     |
| s2       | XmString | read   | value     |

---

## XmStringConcat

### FORMAT

result XmStringConcat  
(s1, s2)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| s1       | XmString | read   | value     |
| s2       | XmString | read   | value     |

---

## XmStringCopy

### FORMAT

result XmStringCopy  
(s1)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| s1       | XmString | read   | value     |

---

## XmStringCreate

### FORMAT

result XmStringCreate  
(text, charset)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| text     | XtString | read   | reference |



| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| charset  | XtString | read   | reference |

---

## XmStringCreateLocalized

### FORMAT

result XmStringCreateLocalized  
(text)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| text     | XtString | read   | reference |

---

## XmStringCreateLtoR

### FORMAT

result XmStringCreateLtoR  
(text, charset)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| text     | XtString | read   | reference |
| charset  | XtString | read   | reference |

---

## XmStringCreateSimple

### FORMAT

result XmStringCreateSimple  
(text)

## OSF/Motif Toolkit Routines

### XmStringCreateSimple

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| text     | XtString | read   | reference |

---

## XmStringDirectionCreate

### FORMAT

result XmStringDirectionCreate  
(direction)

#### Argument Information

| Argument  | Type              | Access | Mechanism |
|-----------|-------------------|--------|-----------|
| result    | XmString          | write  | value     |
| direction | XmStringDirection | read   | value     |

---

## XmStringDraw

### FORMAT

XmStringDraw  
(d, w, fontlist, string, gc, x, y, width, alignment, layout\_direction, clip)

#### Argument Information

| Argument         | Type          | Access | Mechanism |
|------------------|---------------|--------|-----------|
| d                | Display       | read   | value     |
| w                | Window        | read   | value     |
| fontlist         | XmFontList    | read   | value     |
| string           | XmString      | read   | value     |
| gc               | GC            | read   | value     |
| x                | Position      | read   | value     |
| y                | Position      | read   | value     |
| width            | Dimension     | read   | value     |
| alignment        | BYTE UNSIGNED | read   | value     |
| layout_direction | BYTE UNSIGNED | read   | value     |

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| clip     | XRectangle | read   | reference |

---

## XmStringDrawImage

### FORMAT

XmStringDrawImage

(d, w, fontlist, string, gc, x, y, width, alignment, layout\_direction, clip)

### Argument Information

| Argument         | Type          | Access | Mechanism |
|------------------|---------------|--------|-----------|
| d                | Display       | read   | value     |
| w                | Window        | read   | value     |
| fontlist         | XmFontList    | read   | value     |
| string           | XmString      | read   | value     |
| gc               | GC            | read   | value     |
| x                | Position      | read   | value     |
| y                | Position      | read   | value     |
| width            | Dimension     | read   | value     |
| alignment        | BYTE UNSIGNED | read   | value     |
| layout_direction | BYTE UNSIGNED | read   | value     |
| clip             | XRectangle    | read   | reference |

---

## XmStringDrawUnderline

### FORMAT

XmStringDrawUnderline

(d, w, fontlist, string, gc, x, y, width, alignment, layout\_direction, clip, underline)

### Argument Information

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| d        | Display    | read   | value     |
| w        | Window     | read   | value     |
| fontlist | XmFontList | read   | value     |
| string   | XmString   | read   | value     |

## OSF/Motif Toolkit Routines

### XmStringDrawUnderline

| Argument         | Type          | Access | Mechanism |
|------------------|---------------|--------|-----------|
| gc               | GC            | read   | value     |
| x                | Position      | read   | value     |
| y                | Position      | read   | value     |
| width            | Dimension     | read   | value     |
| alignment        | BYTE UNSIGNED | read   | value     |
| layout_direction | BYTE UNSIGNED | read   | value     |
| clip             | XRectangle    | read   | reference |
| underline        | XmString      | read   | value     |

---

## XmStringEmpty

### FORMAT

result XmStringEmpty  
(s1)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| s1       | XmString | read   | value     |

---

## XmStringExtent

### FORMAT

XmStringExtent  
(fontlist, text, width, height)

### Argument Information

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| fontlist | XmFontList | read   | value     |
| text     | XmString   | read   | value     |
| width    | Dimension  | write  | reference |
| height   | Dimension  | write  | reference |

---

## XmStringFree

### FORMAT

XmStringFree  
(string)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| string   | XmString | read   | value     |

---

## XmStringFreeContext

### FORMAT

result XmStringFreeContext  
(context)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmString        | write  | value     |
| context  | XmStringContext | read   | value     |

---

## XmStringGetLtoR

### FORMAT

result XmStringGetLtoR  
(s1, charset, text)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| s1       | XmString | read   | value     |
| charset  | XtString | read   | reference |
| text     | ADDRESS  | write  | reference |

---

## XmStringGetComponent

### FORMAT

result XmStringGetComponent  
(context, text, charset, direction, unknown\_tag, unknown\_length, unknown\_value)

### Argument Information

| Argument       | Type                    | Access | Mechanism |
|----------------|-------------------------|--------|-----------|
| result         | XmStringComponentType   | write  | value     |
| context        | XmStringContext         | read   | value     |
| text           | ADDRESS                 | write  | reference |
| charset        | XmStringCharSet         | write  | reference |
| direction      | XmStringDirection       | write  | reference |
| unknown_tag    | XmStringComponentType   | write  | reference |
| unknown_length | WORD UNSIGNED           | write  | reference |
| unknown_value  | ADDRESS (BYTE UNSIGNED) | write  | reference |

---

## XmStringGetNextSegment

### FORMAT

result XmStringGetNextSegment  
(context, text, charset, direction, separator)

### Argument Information

| Argument  | Type              | Access | Mechanism |
|-----------|-------------------|--------|-----------|
| result    | Boolean           | write  | value     |
| context   | XmStringContext   | read   | value     |
| text      | ADDRESS           | write  | reference |
| charset   | XmStringCharSet   | write  | reference |
| direction | XmStringDirection | write  | reference |
| separator | Boolean           | write  | reference |

---

## XmStringHasSubstring

### FORMAT

result XmStringHasSubstring  
(string, substring)

### Argument Information

| Argument  | Type     | Access | Mechanism |
|-----------|----------|--------|-----------|
| result    | Boolean  | write  | value     |
| string    | XmString | read   | value     |
| substring | XmString | read   | value     |

---

## XmStringHeight

### FORMAT

result XmStringHeight  
(fontlist, string)

### Argument Information

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| result   | Dimension  | write  | value     |
| fontlist | XmFontList | read   | value     |
| string   | XmString   | read   | value     |

---

## XmStringInitContext

### FORMAT

result XmStringInitContext  
(context, string)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |

## OSF/Motif Toolkit Routines

### XmStringInitContext

---

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| context  | XmStringContext | write  | reference |
| string   | XmString        | read   | value     |

---

### XmStringLength

#### FORMAT

result XmStringLength  
(s1)

#### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| s1       | XmString | read   | value     |

---

### XmStringLineCount

#### FORMAT

result XmStringLineCount  
(string)

#### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| string   | XmString | read   | value     |

---

### XmStringNConcat

#### FORMAT

result XmStringNConcat  
(s1, s2, num\_bytes)



**Argument Information**

| Argument  | Type     | Access | Mechanism |
|-----------|----------|--------|-----------|
| result    | XmString | write  | value     |
| s1        | XmString | read   | value     |
| s2        | XmString | read   | value     |
| num_bytes | LONGWORD | read   | value     |

---

**XmStringNCopy**

**FORMAT**

result XmStringNCopy  
(s1, num\_bytes)

**Argument Information**

| Argument  | Type     | Access | Mechanism |
|-----------|----------|--------|-----------|
| result    | XmString | write  | value     |
| s1        | XmString | read   | value     |
| num_bytes | LONGWORD | read   | value     |

---

**XmStringPeekNextComponent**

**FORMAT**

result XmStringPeekNextComponent  
(context)

**Argument Information**

| Argument | Type                  | Access | Mechanism |
|----------|-----------------------|--------|-----------|
| result   | XmStringComponentType | write  | value     |
| context  | XmStringContext       | read   | value     |

---

## XmStringSegmentCreate

### FORMAT

result XmStringSegmentCreate  
(text, charset, direction, separator)

### Argument Information

| Argument  | Type              | Access | Mechanism |
|-----------|-------------------|--------|-----------|
| result    | XmString          | write  | value     |
| text      | XtString          | read   | reference |
| charset   | XtString          | read   | reference |
| direction | XmStringDirection | read   | value     |
| separator | Boolean           | read   | value     |

---

## XmStringSeparatorCreate

### FORMAT

result XmStringSeparatorCreate  
( )

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |

---

## XmStringWidth

### FORMAT

result XmStringWidth  
(fontlist, string)

### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| result   | Dimension | write  | value     |

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| fontlist | XmFontList | read   | value     |
| string   | XmString   | read   | value     |

---

## XmTargetsAreCompatible

### FORMAT

result XmTargetsAreCompatible  
(dpy, exportTargets, numExportTargets, importTargets, numImportTargets)

### Argument Information

| Argument         | Type     | Access | Mechanism |
|------------------|----------|--------|-----------|
| result           | Boolean  | write  | value     |
| dpy              | Display  | read   | reference |
| exportTargets    | Atom     | read   | reference |
| numExportTargets | Cardinal | read   | value     |
| importTargets    | Atom     | read   | reference |
| numImportTargets | Cardinal | read   | value     |

---

## XmTextClearSelection

### FORMAT

XmTextClearSelection  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

## XmTextCopy

### FORMAT

result XmTextCopy  
(widget, time)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |
| time     | Time    | read   | value     |

---

## XmTextCut

### FORMAT

result XmTextCut  
(widget, time)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |
| time     | Time    | read   | value     |

---

## XmTextDisableRedisplay

### FORMAT

XmTextDisableRedisplay  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

---

## XmTextEnableRedisplay

### FORMAT

XmTextEnableRedisplay  
(widget)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

## XmTextFieldClearSelection

### FORMAT

XmTextFieldClearSelection  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

---

## XmTextFieldCopy

### FORMAT

XmTextFieldCopy  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

## XmTextFieldCut

### FORMAT

result XmTextFieldCut  
(widget, time )

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |
| time     | Time    | read   | value     |

---

## XmTextFieldGetBaseline

### FORMAT

result XmTextFieldGetBaseline  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextFieldGetCursorPosition

### FORMAT

result XmTextFieldGetCursorPosition  
(widget)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

---

## XmTextFieldGetEditable

### FORMAT

result XmTextFieldGetEditable  
(widget)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

---

## XmTextFieldGetInsertionPosition

### FORMAT

result XmTextFieldGetInsertionPosition  
(widget)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

## XmTextFieldGetLastPosition

### FORMAT

result XmTextFieldGetLastPosition  
(widget )

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

## XmTextFieldGetMaxLength

### FORMAT

result XmTextFieldGetMaxLength  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextFieldGetSelection

### FORMAT

result XmTextFieldGetSelection  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextFieldGetSelectionPosition

### FORMAT

result XmTextFieldGetSelectionPosition  
(widget, left, right)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Boolean        | write  | value     |
| widget   | Widget         | read   | value     |
| left     | XmTextPosition | write  | reference |



---

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| right    | XmTextPosition | write  | reference |

---

---

## XmTextFieldGetSelectionWcs

### FORMAT

result XmTextFieldGetSelectionWcs  
(widget)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

---

## XmTextFieldGetString

### FORMAT

XtString XmTextFieldGetString  
(widget)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| XtString | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

---

## XmTextFieldGetStringWcs

### FORMAT

result XmTextFieldGetStringWcs  
(widget)

## OSF/Motif Toolkit Routines

### XmTextFieldGetStringWcs

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextFieldGetSubstring

### FORMAT

result XmTextFieldGetSubstring  
(widget, start, num\_chars, buf\_size, buffer)

#### Argument Information

| Argument  | Type           | Access | Mechanism |
|-----------|----------------|--------|-----------|
| result    | LONGWORD       | write  | value     |
| widget    | Widget         | read   | value     |
| start     | XmTextPosition | read   | value     |
| num_chars | LONGWORD       | read   | value     |
| buf_size  | LONGWORD       | read   | value     |
| buffer    | XtString       | read   | reference |

---

## XmTextFieldGetSubstringWcs

### FORMAT

result XmTextFieldGetSubstringWcs  
(widget, start, num\_chars, buf\_size, buffer)

#### Argument Information

| Argument  | Type           | Access | Mechanism |
|-----------|----------------|--------|-----------|
| result    | LONGWORD       | write  | value     |
| widget    | Widget         | read   | value     |
| start     | XmTextPosition | read   | value     |
| num_chars | LONGWORD       | read   | value     |
| buf_size  | LONGWORD       | read   | value     |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| buffer   | XtString | read   | reference |

---

## XmTextFieldInsert

### FORMAT

XmTextFieldInsert  
(widget, position, value)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |
| value    | XtString       | read   | reference |

---

## XmTextFieldInsertWcs

### FORMAT

XmTextFieldInsertWcs  
(widget, position, wcstring)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |
| wcstring | XtString       | read   | reference |

---

## XmTextFieldPaste

### FORMAT

result XmTextFieldPaste  
(widget)

## OSF/Motif Toolkit Routines

### XmTextFieldPaste

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

---

## XmTextFieldPosToXY

### FORMAT

result XmTextFieldPosToXY  
(widget, position, x, y)

#### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Boolean        | write  | value     |
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |
| x        | Position       | write  | reference |
| y        | Position       | write  | reference |

---

## XmTextFieldRemove

### FORMAT

result XmTextFieldRemove  
(widget)

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

---

## XmTextFieldReplace

### FORMAT

XmTextFieldReplace  
(widget, from\_pos, to\_pos, value)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| from_pos | XmTextPosition | read   | value     |
| to_pos   | XmTextPosition | read   | value     |
| value    | XtString       | read   | reference |

---

## XmTextFieldReplaceWcs

### FORMAT

XmTextFieldReplaceWcs  
(widget, from\_pos, to\_pos, wc\_value)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| from_pos | XmTextPosition | read   | value     |
| to_pos   | XmTextPosition | read   | value     |
| wc_value | XtString       | read   | reference |

---

## XmTextFieldSetAddMode

### FORMAT

XmTextFieldSetAddMode  
(widget, state)

## OSF/Motif Toolkit Routines

### XmTextFieldSetAddMode

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| state    | Boolean | read   | value     |

---

## XmTextFieldSetCursorPosition

### FORMAT

XmTextFieldSetCursorPosition  
(widget, position)

#### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |

---

## XmTextFieldSetEditable

### FORMAT

XmTextFieldSetEditable  
(widget, editable)

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| editable | Boolean | read   | value     |

---

## XmTextFieldSetHighlight

### FORMAT

XmTextFieldSetHighlight  
(widget, left, right, mode)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | Widget          | read   | value     |
| left     | XmTextPosition  | read   | value     |
| right    | XmTextPosition  | read   | value     |
| mode     | XmHighlightMode | read   | value     |

---

## XmTextFieldSetInsertionPosition

### FORMAT

XmTextFieldSetInsertionPosition  
(widget, position)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |

---

## XmTextFieldSetMaxLength

### FORMAT

XmTextFieldSetMaxLength  
(widget, max\_length)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

## OSF/Motif Toolkit Routines

### XmTextFieldSetMaxLength

---

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| max_length | LONGWORD | read   | value     |

---

---

## XmTextFieldSetSelection

### FORMAT

XmTextFieldSetSelection  
(widget, first, last, time)

### Argument Information

---

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| first    | XmTextPosition | read   | value     |
| last     | XmTextPosition | read   | value     |
| time     | Time           | read   | value     |

---

---

## XmTextFieldSetString

### FORMAT

XmTextFieldSetString  
(widget, value)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| value    | XtString | read   | reference |

---

---

## XmTextFieldShowPosition

### FORMAT

XmTextFieldShowPosition  
(widget, position)



**Argument Information**

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |

---

**XmTextFieldXYToPos**

**FORMAT**

result XmTextFieldXYToPos  
(widget, x, y)

**Argument Information**

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |
| x        | Position       | read   | value     |
| y        | Position       | read   | value     |

---

**XmTextFindString**

**FORMAT**

result XmTextFindString  
(widget, start, search\_string, direction, position)

**Argument Information**

| Argument      | Type            | Access | Mechanism |
|---------------|-----------------|--------|-----------|
| result        | Boolean         | write  | value     |
| widget        | Widget          | read   | value     |
| start         | XmTextPosition  | read   | value     |
| search_string | XtString        | read   | reference |
| direction     | XmTextDirection | read   | value     |
| position      | XmTextPosition  | write  | reference |

## XmTextFindStringWcs

### FORMAT

result XmTextFindStringWcs  
(widget, start, wc\_string, direction, position )

### Argument Information

| Argument  | Type            | Access | Mechanism |
|-----------|-----------------|--------|-----------|
| result    | Boolean         | write  | value     |
| widget    | Widget          | read   | value     |
| start     | XmTextPosition  | read   | value     |
| wc_string | XtString        | read   | reference |
| direction | XmTextDirection | read   | value     |
| position  | XmTextPosition  | write  | reference |

---

## XmTextGetBaseline

### FORMAT

result XmTextGetBaseline  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextGetCursorPosition

### FORMAT

result XmTextGetCursorPosition  
(widget)

Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

## XmTextGetEditable

### FORMAT

result XmTextGetEditable  
(widget)

Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

---

## XmTextGetInsertionPosition

### FORMAT

result XmTextGetInsertionPosition  
(widget)

Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

## XmTextGetLastPosition

### FORMAT

result XmTextGetLastPosition  
(widget)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

## XmTextGetMaxLength

### FORMAT

result XmTextGetMaxLength  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextGetSelection

### FORMAT

result XmTextGetSelection  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextGetSelectionPosition

### FORMAT

result XmTextGetSelectionPosition  
(widget, left, right)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Boolean        | write  | value     |
| widget   | Widget         | read   | value     |
| left     | XmTextPosition | write  | reference |
| right    | XmTextPosition | write  | reference |

---

## XmTextGetSelectionWcs

### FORMAT

result XmTextGetSelectionWcs  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

## XmTextGetSource

### FORMAT

result XmTextGetSource  
(widget)

### Argument Information

| Argument | Type         | Access | Mechanism |
|----------|--------------|--------|-----------|
| result   | XmTextSource | write  | value     |

## OSF/Motif Toolkit Routines

### XmTextGetSource

---

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

---

## XmTextGetString

### FORMAT

result XmTextGetString  
(widget)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

---

## XmTextGetStringWcs

### FORMAT

result XmTextGetStringWcs  
(widget)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| widget   | Widget   | read   | value     |

---

---

## XmTextGetSubstring

### FORMAT

result XmTextGetSubstring  
(widget, start, num\_chars, buf\_size, buffer)

**Argument Information**

| Argument  | Type           | Access | Mechanism |
|-----------|----------------|--------|-----------|
| result    | LONGWORD       | write  | value     |
| widget    | Widget         | read   | value     |
| start     | XmTextPosition | read   | value     |
| num_chars | LONGWORD       | read   | value     |
| buf_size  | LONGWORD       | read   | value     |
| buffer    | XtString       | read   | reference |

---

**XmTextGetSubstringWcs**

**FORMAT**

result XmTextGetSubstringWcs  
(widget, start, num\_chars, buf\_size, buffer)

**Argument Information**

| Argument  | Type           | Access | Mechanism |
|-----------|----------------|--------|-----------|
| result    | LONGWORD       | write  | value     |
| widget    | Widget         | read   | value     |
| start     | XmTextPosition | read   | value     |
| num_chars | LONGWORD       | read   | value     |
| buf_size  | LONGWORD       | read   | value     |
| buffer    | XtString       | read   | reference |

---

**XmTextGetTopCharacter**

**FORMAT**

result XmTextGetTopCharacter  
(widget)

**Argument Information**

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | XmTextPosition | write  | value     |
| widget   | Widget         | read   | value     |

---

## XmTextInsert

### FORMAT

XmTextInsert  
(widget, position, value)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |
| value    | XtString       | read   | reference |

---

## XmTextInsertWcs

### FORMAT

XmTextInsertWcs  
(widget, position, wc\_value)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |
| wc_value | XtString       | read   | reference |

---

## XmTextPaste

### FORMAT

result XmTextPaste  
(widget)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |



| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

## XmTextPosToXY

### FORMAT

result XmTextPosToXY  
(widget, position, x, y)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Boolean        | write  | value     |
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |
| x        | Position       | write  | reference |
| y        | Position       | write  | reference |

---

## XmTextRemove

### FORMAT

result XmTextRemove  
(widget)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

## XmTextReplace

### FORMAT

XmTextReplace  
(widget, from\_pos, to\_pos, value )

### Argument Information

---

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | write  | value     |
| from_pos | XmTextPosition | read   | value     |
| to_pos   | XmTextPosition | read   | value     |
| value    | XtString       | read   | reference |

---

## XmTextReplaceWcs

### FORMAT

XmTextReplaceWcs  
(widget, from\_pos, to\_pos, value)

### Argument Information

---

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| from_pos | XmTextPosition | read   | value     |
| to_pos   | XmTextPosition | read   | value     |
| value    | XtString       | read   | reference |

---

## XmTextScroll

### FORMAT

XmTextScroll  
(widget, lines)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| lines    | LONGWORD | read   | value     |

---

**XmTextSetAddMode**

**FORMAT**

XmTextSetAddMode  
(widget, state)

**Argument Information**

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| state    | Boolean | read   | value     |

---

**XmTextSetCursorPosition**

**FORMAT**

XmTextSetCursorPosition  
(widget, position)

**Argument Information**

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |

---

## XmTextSetEditable

### FORMAT

XmTextSetEditable  
(widget, editable)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| editable | Boolean | read   | value     |

---

## XmTextSetHighlight

### FORMAT

XmTextSetHighlight  
(widget, left, right, mode)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | Widget          | read   | value     |
| left     | XmTextPosition  | read   | value     |
| right    | XmTextPosition  | read   | value     |
| mode     | XmHighlightMode | read   | value     |

---

## XmTextSetInsertionPosition

### FORMAT

XmTextSetInsertionPosition  
(widget, position)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| position | XmTextPosition | read   | value     |

---

## XmTextSetMaxLength

### FORMAT

XmTextSetMaxLength  
(widget, max\_length)

### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| widget     | Widget   | read   | value     |
| max_length | LONGWORD | read   | value     |

---

## XmTextSetSelection

### FORMAT

XmTextSetSelection  
(widget, first, last, time)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| first    | XmTextPosition | read   | value     |
| last     | XmTextPosition | read   | value     |
| time     | Time           | read   | value     |

---

## XmTextSetSource

### FORMAT

XmTextSetSource  
(widget, source, top\_character, cursor\_position)

## OSF/Motif Toolkit Routines

### XmTextSetSource

#### Argument Information

| Argument        | Type           | Access | Mechanism |
|-----------------|----------------|--------|-----------|
| widget          | Widget         | read   | value     |
| source          | XmTextSource   | read   | value     |
| top_character   | XmTextPosition | read   | value     |
| cursor_position | XmTextPosition | read   | value     |

---

## XmTextSetString

#### FORMAT

XmTextSetString  
(widget, value)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| value    | XtString | read   | reference |

---

## XmTextSetStringWcs

#### FORMAT

XmTextSetStringWcs  
(widget, wc\_value)

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| wc_value | XtString | read   | reference |

---

## XmTextSetTopCharacter

### FORMAT

XmTextSetTopCharacter  
(widget, top\_character)

### Argument Information

| Argument      | Type           | Access | Mechanism |
|---------------|----------------|--------|-----------|
| widget        | Widget         | read   | value     |
| top_character | XmTextPosition | read   | value     |

---

## XmTextShowPosition

### FORMAT

XmTextShowPosition  
(widget, position)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| widget   | Widget         | read   | value     |
| position | XmTextPosition | read   | value     |

---

## XmTextXYToPos

### FORMAT

XmTextPosition XmTextXYToPos  
(widget, x, y)

### Argument Information

| Argument       | Type           | Access | Mechanism |
|----------------|----------------|--------|-----------|
| XmTextPosition | XmTextPosition | write  | value     |
| widget         | Widget         | read   | value     |
| x              | Position       | read   | value     |

## OSF/Motif Toolkit Routines

### XmTextXYToPos

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| y        | Position | read   | value     |

---

---

## XmToggleButtonGadgetGetState

### FORMAT

result XmToggleButtonGadgetGetState  
(widget)

### Argument Information

---

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

---

---

## XmToggleButtonGadgetSetState

### FORMAT

XmToggleButtonGadgetSetState  
(widget, state, notify)

### Argument Information

---

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| state    | Boolean | read   | value     |
| notify   | Boolean | read   | value     |

---

---

## XmToggleButtonGetState

### FORMAT

result XmToggleButtonGetState  
(widget)



**Argument Information**

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |

---

**XmToggleButtonSetState**

**FORMAT**

XmToggleButtonSetState  
(widget, state, notify)

**Argument Information**

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| state    | Boolean | read   | value     |
| notify   | Boolean | read   | value     |

---

**XmTrackingEvent**

**FORMAT**

result XmTrackingEvent  
(widget, cursor, confine\_to, event)

**Argument Information**

| Argument   | Type             | Access | Mechanism |
|------------|------------------|--------|-----------|
| result     | Widget           | write  | value     |
| widget     | Widget           | read   | value     |
| cursor     | Cursor           | read   | value     |
| confine_to | Boolean          | read   | value     |
| event      | XKeyPressedEvent | read   | reference |

## OSF/Motif Toolkit Routines

### XmTrackingLocate

---

## XmTrackingLocate

### FORMAT

result XmTrackingLocate  
(widget, cursor, confine\_to)

### Argument Information

| Argument   | Type    | Access | Mechanism |
|------------|---------|--------|-----------|
| result     | Widget  | write  | value     |
| widget     | Widget  | read   | value     |
| cursor     | Cursor  | read   | value     |
| confine_to | Boolean | read   | value     |

---

## XmTranslateKey

### FORMAT

XmTranslateKey  
(dpy, keycode, modifiers, modifiers\_return, keysym\_return)

### Argument Information

| Argument         | Type      | Access | Mechanism |
|------------------|-----------|--------|-----------|
| dpy              | Display   | read   | value     |
| keycode          | KeyCode   | read   | value     |
| modifiers        | Modifiers | read   | value     |
| modifiers_return | Modifiers | write  | reference |
| keysym_return    | KeySym    | write  | reference |

---

## XmUninstallImage

### FORMAT

result XmUninstallImage  
(image)

**Argument Information**

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| image    | XImage  | read   | reference |

---

**XmUpdateDisplay**

**FORMAT**

XmUpdateDisplay  
(widget)

**Argument Information**

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

**XmVaCreateSimpleCheckBox**

**FORMAT**

result XmVaCreateSimpleCheckBox  
(parent, name, callback)

**Argument Information**

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Widget         | write  | value     |
| parent   | Widget         | read   | value     |
| name     | XtString       | read   | reference |
| callback | XtCallbackProc | read   | value     |

---

## XmVaCreateSimpleMenuBar

### FORMAT

result XmVaCreateSimpleMenuBar  
(parent, name)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |

---

## XmVaCreateSimpleOptionMenu

### FORMAT

result XmVaCreateSimpleOptionMenu  
(parent, name, option\_label, option\_mnemonic, button\_set, callback)

### Argument Information

| Argument        | Type           | Access | Mechanism |
|-----------------|----------------|--------|-----------|
| result          | Widget         | write  | value     |
| parent          | Widget         | read   | value     |
| name            | XtString       | read   | reference |
| option_label    | XmString       | read   | value     |
| option_mnemonic | KeySym         | read   | value     |
| button_set      | LONGWORD       | read   | value     |
| callback        | XtCallbackProc | read   | value     |

---

## XmVaCreateSimplePopupMenu

### FORMAT

result XmVaCreateSimplePopupMenu  
(parent, name, callback)

Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Widget         | write  | value     |
| parent   | Widget         | read   | value     |
| name     | XtString       | read   | reference |
| callback | XtCallbackProc | read   | value     |

---

## XmVaCreateSimplePulldownMenu

**FORMAT**

result XmVaCreateSimplePulldownMenu  
(parent, name, post\_from\_button, callback)

Argument Information

| Argument         | Type           | Access | Mechanism |
|------------------|----------------|--------|-----------|
| result           | Widget         | write  | value     |
| parent           | Widget         | read   | value     |
| name             | XtString       | read   | reference |
| post_from_button | LONGWORD       | read   | value     |
| callback         | XtCallbackProc | read   | value     |

---

## XmVaCreateSimpleRadioBox

**FORMAT**

result XmVaCreateSimpleRadioBox  
(parent, name, button\_set, callback)

Argument Information

| Argument   | Type           | Access | Mechanism |
|------------|----------------|--------|-----------|
| result     | Widget         | write  | value     |
| parent     | Widget         | read   | value     |
| name       | XtString       | read   | reference |
| button_set | LONGWORD       | read   | value     |
| callback   | XtCallbackProc | read   | value     |

---

## XmWidgetGetBaselines

### FORMAT

result XmWidgetGetBaselines  
(wid, baselines, line\_count)

### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | Boolean  | write  | value     |
| wid        | Widget   | read   | value     |
| baselines  | ADDRESS  | write  | reference |
| line_count | LONGWORD | write  | reference |

---

## XmWidgetGetDisplayRect

### FORMAT

result XmWidgetGetDisplayRect  
(wid, displayrect)

### Argument Information

| Argument    | Type       | Access | Mechanism |
|-------------|------------|--------|-----------|
| result      | Boolean    | write  | value     |
| wid         | Widget     | read   | value     |
| displayrect | XRectangle | read   | reference |

---

## OSF/Motif Toolkit Data Structures

This chapter documents OSF/Motif toolkit data structures.

---

### MotifWmHints

#### Field Information

---

| Argument    | Type     |
|-------------|----------|
| flags       | LONGWORD |
| functions   | LONGWORD |
| decorations | LONGWORD |
| input_mode  | LONGWORD |
| status      | LONGWORD |

---

---

### MotifWmInfo

#### Field Information

---

| Argument  | Type     |
|-----------|----------|
| flags     | LONGWORD |
| wm_window | ADDRESS  |

---

---

### MrmOsOpenParam

#### Field Information

---

| Argument      | Type     |
|---------------|----------|
| version       | Cardinal |
| default_fname | XtString |
| nam_flg       | Union    |

---

| Argument    | Type              |
|-------------|-------------------|
| related_nam | LONGWORD UNSIGNED |
| clobber_flg | Boolean           |
| display     | ADDRESS           |

---

## MrmRegisterArg

### Field Information

| Argument | Type    |
|----------|---------|
| name     | ADDRESS |
| value    | Opaque  |

---

## PropMotifWmHints

### Field Information

| Argument    | Type     |
|-------------|----------|
| flags       | LONGWORD |
| functions   | LONGWORD |
| decorations | LONGWORD |
| inputMode   | LONGWORD |
| status      | LONGWORD |

---

## PropMotifWmInfo

### Field Information

| Argument | Type     |
|----------|----------|
| flags    | LONGWORD |
| wmWindow | ADDRESS  |



---

## XmAnyCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |

---

---

## XmAnyICCCallbackStruct

### Field Information

---

| Argument  | Type     |
|-----------|----------|
| reason    | LONGWORD |
| event     | ADDRESS  |
| timestamp | Time     |

---

---

## XmArrowButtonCallbackStruct

### Field Information

---

| Argument    | Type     |
|-------------|----------|
| reason      | LONGWORD |
| event       | ADDRESS  |
| click_count | LONGWORD |

---

---

## XmClipboardPendingRec

### Field Information

---

| Argument  | Type     |
|-----------|----------|
| DataId    | LONGWORD |
| PrivateId | LONGWORD |

---

---

## XmCommandCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |
| value    | XmString |
| length   | LONGWORD |

---

---

## XmDragDropFinishCallbackStruct

### Field Information

---

| Argument  | Type     |
|-----------|----------|
| reason    | LONGWORD |
| event     | ADDRESS  |
| timestamp | Time     |

---

---

## XmDropProcCallbackStruct

### Field Information

---

| Argument       | Type          |
|----------------|---------------|
| reason         | LONGWORD      |
| event          | ADDRESS       |
| timestamp      | Time          |
| dragContext    | Widget        |
| x              | Position      |
| y              | Position      |
| dropSiteStatus | BYTE UNSIGNED |
| operation      | BYTE UNSIGNED |
| operations     | BYTE UNSIGNED |
| dropAction     | BYTE UNSIGNED |

---

---

## XmDragMotionCallbackStruct

### Field Information

---

| Argument       | Type          |
|----------------|---------------|
| reason         | LONGWORD      |
| event          | ADDRESS       |
| timestamp      | Time          |
| operation      | BYTE UNSIGNED |
| operations     | BYTE UNSIGNED |
| dropSiteStatus | BYTE UNSIGNED |
| x              | Position      |
| y              | Position      |

---

---

## XmDragProcCallbackStruct

### Field Information

---

| Argument    | Type          |
|-------------|---------------|
| reason      | LONGWORD      |
| event       | ADDRESS       |
| timestamp   | Time          |
| dragContext | Widget        |
| operation   | BYTE UNSIGNED |
| operations  | BYTE UNSIGNED |
| animate     | Boolean       |

---

---

## XmDrawingAreaCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |
| window   | ADDRESS  |

---

## OSF/Motif Toolkit Data Structures

### XmDrawnButtonCallbackStruct

---

## XmDrawnButtonCallbackStruct

### Field Information

---

| Argument    | Type     |
|-------------|----------|
| reason      | LONGWORD |
| event       | ADDRESS  |
| window      | ADDRESS  |
| click_count | LONGWORD |

---

---

## XmDropFinishCallbackStruct

### Field Information

---

| Argument         | Type          |
|------------------|---------------|
| reason           | LONGWORD      |
| event            | ADDRESS       |
| timestamp        | Time          |
| operation        | BYTE UNSIGNED |
| operations       | BYTE UNSIGNED |
| dropSiteStatus   | BYTE UNSIGNED |
| dropAction       | BYTE UNSIGNED |
| completionStatus | BYTE UNSIGNED |

---

---

## XmDropSiteEnterCallbackStruct

### Field Information

---

| Argument       | Type          |
|----------------|---------------|
| reason         | LONGWORD      |
| event          | ADDRESS       |
| timestamp      | Time          |
| operation      | BYTE UNSIGNED |
| operations     | BYTE UNSIGNED |
| dropSiteStatus | BYTE UNSIGNED |
| x              | Position      |

---

---

| Argument | Type     |
|----------|----------|
| y        | Position |

---

---

## XmDropSiteLeaveCallbackStruct

### Field Information

---

| Argument  | Type     |
|-----------|----------|
| reason    | LONGWORD |
| event     | ADDRESS  |
| timestamp | Time     |

---

---

## XmDropSiteVisualsRec

### Field Information

---

| Argument           | Type      |
|--------------------|-----------|
| background         | Pixel     |
| foreground         | Pixel     |
| topShadowColor     | Pixel     |
| topShadowPixmap    | Pixmap    |
| bottomShadowColor  | Pixel     |
| bottomShadowPixmap | Pixmap    |
| shadowThickness    | Dimension |
| highlightColor     | Pixel     |
| highlightPixmap    | Pixmap    |
| highlightThickness | Dimension |
| borderWidth        | Dimension |

---

---

## XmDropStartCallbackStruct

### Field Information

---

| Argument       | Type          |
|----------------|---------------|
| reason         | LONGWORD      |
| event          | ADDRESS       |
| timestamp      | Time          |
| operation      | BYTE UNSIGNED |
| operations     | BYTE UNSIGNED |
| dropSiteStatus | BYTE UNSIGNED |
| dropAction     | BYTE UNSIGNED |
| x              | Position      |
| y              | Position      |
| window         | ADDRESS       |
| iccHandle      | Atom          |

---

---

## XmDropTransferEntryRec

### Field Information

---

| Argument    | Type      |
|-------------|-----------|
| client_data | XtPointer |
| target      | ADDRESS   |

---

---

## XmFileSelectionCallbackStruct

### Field Information

---

| Argument    | Type     |
|-------------|----------|
| reason      | LONGWORD |
| event       | ADDRESS  |
| value       | XmString |
| length      | LONGWORD |
| mask        | XmString |
| mask_length | LONGWORD |

---

## OSF/Motif Toolkit Data Structures XmFileSelectionCallbackStruct

---

| Argument       | Type     |
|----------------|----------|
| dir            | XmString |
| dir_length     | LONGWORD |
| pattern        | XmString |
| pattern_length | LONGWORD |

---

---

## XmListCallbackStruct

### Field Information

---

| Argument                | Type              |
|-------------------------|-------------------|
| reason                  | LONGWORD          |
| event                   | ADDRESS           |
| item                    | XmString          |
| item_length             | LONGWORD          |
| item_position           | LONGWORD          |
| selected_items          | XmStringTable     |
| selected_items_count    | LONGWORD          |
| selected_item_positions | ADDRESS(LONGWORD) |
| selection_type          | BYTE              |

---

---

## XmOperationChangedCallbackStruct

### Field Information

---

| Argument       | Type          |
|----------------|---------------|
| reason         | LONGWORD      |
| event          | ADDRESS       |
| timestamp      | Time          |
| operation      | BYTE UNSIGNED |
| operations     | BYTE UNSIGNED |
| dropSiteStatus | BYTE UNSIGNED |

---

---

## XmPushButtonCallbackStruct

### Field Information

---

| Argument    | Type     |
|-------------|----------|
| reason      | LONGWORD |
| event       | ADDRESS  |
| click_count | LONGWORD |

---

---

## XmRepTypeEntryRec

### Field Information

---

| Argument          | Type          |
|-------------------|---------------|
| rep_type_name     | XtString      |
| value_names       | XtString      |
| values            | ADDRESS       |
| num_values        | BYTE UNSIGNED |
| reverse_installed | Boolean       |
| rep_type_id       | WORD UNSIGNED |

---

---

## XmRepTypeListRec

### Field Information

---

| Argument          | Type          |
|-------------------|---------------|
| rep_type_name     | XtString      |
| value_names       | XtString      |
| values            | ADDRESS       |
| num_values        | BYTE UNSIGNED |
| reverse_installed | Boolean       |
| rep_type_id       | WORD UNSIGNED |

---



---

## XmRowColumnCallbackStruct

### Field Information

---

| Argument       | Type     |
|----------------|----------|
| reason         | LONGWORD |
| event          | ADDRESS  |
| wid            | Widget   |
| data           | ADDRESS  |
| callbackstruct | ADDRESS  |

---

---

## XmScaleCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |
| value    | LONGWORD |

---

---

## XmScrollBarCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |
| value    | LONGWORD |
| pixel    | LONGWORD |

---

## XmSecondaryResourceDataRec

### Field Information

---

| Argument      | Type               |
|---------------|--------------------|
| base_proc     | XmResourceBaseProc |
| client_data   | XtPointer          |
| name          | XtString           |
| res_class     | XtString           |
| resources     | XtResourceList     |
| num_resources | Cardinal           |

---

---

## XmSelectionBoxCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |
| value    | XmString |
| length   | LONGWORD |

---

---

## XmTextBlockRec

### Field Information

---

| Argument | Type         |
|----------|--------------|
| ptr      | ADDRESS      |
| length   | LONGWORD     |
| format   | XmTextFormat |

---

---

## XmTextBlockRecWcs

### Field Information

| Argument | Type     |
|----------|----------|
| wcsptr   | ADDRESS  |
| length   | LONGWORD |

---

## XmTextVerifyCallbackStruct

### Field Information

| Argument   | Type        |
|------------|-------------|
| reason     | LONGWORD    |
| event      | ADDRESS     |
| doit       | Boolean     |
| currInsert | LONGWORD    |
| newInsert  | LONGWORD    |
| startPos   | LONGWORD    |
| endPos     | LONGWORD    |
| text       | XmTextBlock |

---

## XmTextVerifyCallbackStructWcs

### Field Information

| Argument   | Type        |
|------------|-------------|
| reason     | LONGWORD    |
| event      | ADDRESS     |
| doit       | Boolean     |
| currInsert | LONGWORD    |
| newInsert  | LONGWORD    |
| startPos   | LONGWORD    |
| endPos     | LONGWORD    |
| text       | XmTextBlock |

---

## XmToggleButtonCallbackStruct

### Field Information

---

| Argument | Type     |
|----------|----------|
| reason   | LONGWORD |
| event    | ADDRESS  |
| is_set   | LONGWORD |

---

---

## XmTopLevelEnterCallbackStruct

### Field Information

---

| Argument          | Type     |
|-------------------|----------|
| reason            | LONGWORD |
| event             | ADDRESS  |
| timestamp         | Time     |
| screen            | ADDRESS  |
| window            | ADDRESS  |
| x                 | Position |
| y                 | Position |
| dragProtocolStyle | BYTE     |
| iccHandle         | Atom     |

---

---

## XmTopLevelLeaveCallbackStruct

### Field Information

---

| Argument  | Type     |
|-----------|----------|
| reason    | LONGWORD |
| event     | ADDRESS  |
| timestamp | Time     |
| screen    | ADDRESS  |
| window    | ADDRESS  |

---

---

## XmTraverseObscuredCallbackStru

### Field Information

---

| Argument              | Type                 |
|-----------------------|----------------------|
| reason                | LONGWORD             |
| event                 | ADDRESS              |
| traversal_destination | Widget               |
| direction             | XmTraversalDirection |

---



# Part IV

---

## DECwindows Toolkit Extensions

Part IV documents DECwindows Toolkit Extensions routines and data structures and includes the following chapters:

- Chapter 8 – DECwindows Toolkit Extensions Routines
- Chapter 9 – DECwindows Toolkit Extensions Data Structures





---

## DECwindows Toolkit Extensions Routines

This chapter documents DECwindows toolkit extensions routines. See Section 1.2 for information about the format used to describe each routine.

---

### DXmActivateWidget

#### FORMAT

DXmActivateWidget  
(widget)

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |

---

### DXmChangeWindowGeometry

#### FORMAT

DXmChangeWindowGeometry  
(w, size )

#### Argument Information

| Argument | Type             | Access | Mechanism |
|----------|------------------|--------|-----------|
| w        | Widget           | value  | read      |
| size     | XtWidgetGeometry | write  | reference |

## DECwindows Toolkit Extensions Routines

### DXmMakeGeometryRequest

---

## DXmMakeGeometryRequest

### FORMAT

result DXmMakeGeometryRequest  
(w, geom)

### Argument Information

| Argument | Type             | Access | Mechanism |
|----------|------------------|--------|-----------|
| result   | XtGeometryResult | write  | value     |
| w        | Widget           | value  | read      |
| geom     | XtWidgetGeometry | write  | reference |

---

## DXmChildren

### FORMAT

result DXmChildren  
(widget)

### Argument Information

| Argument | Type           | Access | Mechanism |
|----------|----------------|--------|-----------|
| result   | Widget (array) | write  | value     |
| widget   | Widget         | read   | value     |

---

## DXmColorMixGetNewColor

### FORMAT

DXmColorMixGetNewColor  
(cmw, red, green, blue)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| cmw      | DXmColorMixWidget | read   | value     |
| red      | WORD UNSIGNED     | write  | reference |

## DECwindows Toolkit Extensions Routines DXmColorMixGetNewColor

| Argument | Type          | Access | Mechanism |
|----------|---------------|--------|-----------|
| green    | WORD UNSIGNED | write  | reference |
| blue     | WORD UNSIGNED | write  | reference |

---

## DXmColorMixSetNewColor

### FORMAT

DXmColorMixSetNewColor  
(cmw, red, green, blue)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| cmw      | DXmColorMixWidget | read   | value     |
| red      | WORD UNSIGNED     | write  | reference |
| green    | WORD UNSIGNED     | write  | reference |
| blue     | WORD UNSIGNED     | write  | reference |

---

## DXmCreateColorMix

### FORMAT

Widget DXmCreateColorMix  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmCreateColorMixDialog

---

## DXmCreateColorMixDialog

### FORMAT

Widget DXmCreateColorMixDialog  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

---

## DXmCreateCSText

### FORMAT

Widget DXmCreateCSText  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

---

## DXmCreateCursor

### FORMAT

result DXmCreateCursor  
(widget, cursorkind)

**Argument Information**

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | Cursor   | write  | reference |
| widget     | Widget   | read   | value     |
| cursorkind | LONGWORD | read   | value     |

---

## DXmCreateHelp

**FORMAT**

Widget DXmCreateHelp  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | value     |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

---

## DXmCreateHelpDialog

**FORMAT**

Widget DXmCreateHelpDialog  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmCreatePrintBox

---

## DXmCreatePrintBox

### FORMAT

Widget DXmCreatePrintBox  
(parent, name, arglist, widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | Arglist  | read   | reference |
| widget   | Widget   | write  | value     |

---

## DXmCreatePrintDialog

### FORMAT

Widget DXmCreatePrintDialog  
(parent, name, arglist, argcount)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

---

## DXmCreateScrolledCSText

### FORMAT

Widget DXmCSCreateScrolledCSText  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | write  | value     |
| parent   | Widget   | read   | value     |
| name     | XtString | read   | reference |
| arglist  | ArgList  | read   | reference |
| argcount | Cardinal | read   | value     |

---

**DXmCreateSvn**

**FORMAT**

Widget DXmCreateSvn  
(parent, name, arglist, argcount)

**Argument Information**

| Argument | Type        | Access | Mechanism |
|----------|-------------|--------|-----------|
| widget   | Widget      | write  | value     |
| parent   | Widget      | read   | value     |
| name     | XtString    | read   | reference |
| arglist  | Arg (array) | read   | reference |
| argcount | Cardinal    | read   | value     |

---

**DXmCSContainsStringCharSet**

**FORMAT**

result DXmCSContainsStringCharSet  
(str)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Boolean  | write  | value     |
| str      | XmString | read   | value     |

---

## DXmCSTextClearSelection

### FORMAT

DXmCSTextClearSelection  
(widget, time)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| widget   | Widget | read   | value     |
| time     | Time   | read   | value     |

---

## DXmCSTextCopy

### FORMAT

result DXmCSTextCopy  
(widget, time)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |
| time     | Time    | read   | value     |

---

## DXmCSTextCut

### FORMAT

result DXmCSTextCut  
(widget, time)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |



| Argument | Type | Access | Mechanism |
|----------|------|--------|-----------|
| time     | Time | read   | value     |

---

## DXmCSTextDisableRedisplay

### FORMAT

DXmCSTextDisableRedisplay  
(widget, loses\_backing\_store)

### Argument Information

| Argument            | Type            | Access | Mechanism |
|---------------------|-----------------|--------|-----------|
| widget              | DXmCSTextWidget | read   | value     |
| loses_backing_store | Boolean         | write  | value     |

---

## DXmCSTextEnableRedisplay

### FORMAT

DXmCSTextEnableRedisplay  
(widget)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |

---

## DXmCSTextGetEditable

### FORMAT

result DXmCSGetTextEditable  
(widget)

## DECwindows Toolkit Extensions Routines

### DXmCSTextGetEditable

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | Boolean         | write  | value     |
| widget   | DXmCSTextWidget | read   | value     |

---

## DXmCSTextGetInsertionPosition

### FORMAT

result DXmCSTextGetInsertionPosition  
(widget)

#### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCSTextPosition | write  | value     |
| widget   | DXmCSTextWidget   | read   | value     |

---

## DXmCSTextGetLastPosition

### FORMAT

result DXmCSTextGetLastPosition  
(widget)

#### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCsTextPosition | write  | value     |
| widget   | DXmCSTextWidget   | read   | value     |

---

## DXmCSTextGetMaxLength

### FORMAT

result DXmCSTextGetMaxLength  
(widget)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | LONGWORD        | write  | value     |
| widget   | DXmCSTextWidget | read   | value     |

---

## DXmCSTextGetSelection

### FORMAT

result DXmCSTextGetSelection  
(widget)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| widget   | Widget   | read   | value     |

---

## DXmCSTextGetSelectionInfo

### FORMAT

result DXmCSTextGetSelectionInfo  
(widget, left, right)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | Boolean           | write  | reference |
| widget   | Widget            | read   | reference |
| left     | DXmCSTextPosition | modify | reference |

## DECwindows Toolkit Extensions Routines

### DXmCSTextGetSelectionInfo

---

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| right    | DXmCSTextPosition | modify | reference |

---

---

## DXmCSTextGetString

### FORMAT

result DXmCSTextGetString  
(widget)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |
| widget   | Widget   | read   | value     |

---

---

## DXmCSTextGetStringWrapped

### FORMAT

result DXmCSTextGetStringWrapped  
(widget, start, end, result )

### Argument Information

---

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | XmString          | write  | value     |
| widget   | Widget            | read   | value     |
| start    | DXmCSTextPosition | read   | value     |
| end      | DXmCSTextPosition | read   | value     |

---

---

## DXmCSTextGetTextPath

### FORMAT

result DXmCSTextGetTextPath  
(widget)

Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | XmStringDirection | write  | value     |
| widget   | DXmCSTextWidget   | read   | value     |

---

## DXmCSTextGetTopPosition

### FORMAT

result DXmCSTextGetTopPosition  
(widget)

Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCSTextPosition | write  | value     |
| widget   | DXmCSTextWidget   | read   | value     |

---

## DXmCSTextHasSelection

### FORMAT

result DXmCSTextHasSelection  
(widget)

Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | Boolean         | write  | value     |
| widget   | DXmCSTextWidget | read   | value     |

---

## DXmCSTextHorizontalScroll

### FORMAT

DXmCSTextHorizontalScroll  
(widget, n )

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |
| n        | LONGWORD        | read   | value     |

---

## DXmCSTextInsert

### FORMAT

result DXmCSTextInsert  
(widget, pos, value)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCSTextStatus   | write  | value     |
| widget   | Widget            | read   | value     |
| pos      | DXmCSTextPosition | read   | value     |
| value    | XmString          | read   | value     |

---

## DXmCSTextMarkRedraw

### FORMAT

DXmCSTextMarkRedraw  
(widget, left, right)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| left     | DXmCSTextPosition | read   | value     |
| right    | DXmCSTextPosition | read   | value     |

---

## DXmCSTextGetInsertionPosition

### FORMAT

result DXmCSTextGetInsertionPosition  
(widget)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCSTextPosition | write  | reference |
| widget   | DXmCSTextWidget   | read   | value     |

---

## DXmCSTextInvalidate

### FORMAT

DXmCSTextInvalidate  
(widget, left, right)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| widget   | DXmCSTextWidget   | read   | value     |
| left     | DXmCSTextPosition | read   | value     |
| right    | DXmCSTextPosition | read   | value     |

---

## DXmCSTextNumLines

### FORMAT

result DXmCSTextNumLines  
(widget)

## DECwindows Toolkit Extensions Routines

### DXmCSTextNumLines

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | LONGWORD        | write  | value     |
| widget   | DXmCSTextWidget | read   | value     |

---

## DXmCSTextPaste

### FORMAT

result DXmCSTextPaste  
(widget, time)

#### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | Boolean | write  | value     |
| widget   | Widget  | read   | value     |
| time     | Time    | read   | value     |

---

## DXmCSTextPosToLine

### FORMAT

result DXmCSTextPosToLine  
(widget, position )

#### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | LONGWORD          | write  | value     |
| widget   | DXmCSTextWidget   | read   | value     |
| position | DXmCSTextPosition | read   | value     |



---

## DXmCSTextPosToXY

### FORMAT

result DXmCSTextPosToXY  
(widget, position, x, y)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | Boolean           | write  | value     |
| widget   | Widget            | read   | value     |
| position | DXmCSTextPosition | read   | value     |
| x        | Position          | write  | value     |
| y        | Position          | write  | value     |

---

## DXmCSTextRemove

### FORMAT

result DXmCSTextRemove  
(widget)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | Boolean         | write  | value     |
| widget   | DXmCSTextWidget | read   | value     |

---

## DXmCSTextRead

### FORMAT

DXmCSTextRead  
(widget, frompos, topos, value)

## DECwindows Toolkit Extensions Routines

### DXmCSTextRead

#### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| widget   | Widget            | read   | value     |
| frompos  | DXmCSTextPosition | read   | value     |
| topos    | DXmCSTextPosition | read   | value     |
| value    | XmString          | write  | reference |

---

## DXmCSTextReplace

### FORMAT

result DXmCSTextReplace  
(widget, from\_pos, to\_pos, value)

#### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCSTextStatus   | write  | value     |
| widget   | Widget            | read   | value     |
| from_pos | DXmCSTextPosition | read   | value     |
| to_pos   | DXmCSTextPosition | read   | value     |
| value    | XmString          | read   | value     |

---

## DXmCSTextSetAddMode

### FORMAT

DXmCSTextSetAddMode  
(widget, state)

#### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |
| state    | Boolean         | read   | value     |

---

## DXmCSTextSetEditable

### FORMAT

DXmCSTextSetEditable  
(widget, editable)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |
| editable | Boolean         | read   | value     |

---

## DXmCSTextSetHighlight

### FORMAT

DXmCSTextSetHighlight  
(widget, left, right, mode)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| widget   | DXmCSTextWidget   | read   | value     |
| left     | DXmCSTextPosition | read   | value     |
| right    | DXmCSTextPosition | read   | value     |
| mode     | XmHighlightMode   | read   | value     |

---

## DXmCSTextSetInsertionPosition

### FORMAT

DXmCSTextSetInsertionPosition  
(widget, position)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmCSTextSetInsertionPosition

---

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| position | DXmCSTextPosition | read   | value     |

---

---

## DXmCSTextSetMaxLength

### FORMAT

DXmCSTextSetMaxLength  
(widget, max)

### Argument Information

---

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |
| max      | LONGWORD        | read   | value     |

---

---

## DXmCSTextSetSelection

### FORMAT

DXmCSTextSetSelection  
(widget, first, last, time)

### Argument Information

---

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| first    | LONGWORD | read   | value     |
| last     | LONGWORD | read   | value     |
| time     | LONGWORD | read   | value     |

---

---

## DXmCSTextSetString

### FORMAT

DXmCSTextSetString  
(widget,value)

Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| widget   | Widget   | read   | value     |
| value    | XmString | read   | value     |

---

## DXmCSTextSetTopPosition

### FORMAT

DXmCSTextSetTopPosition  
(widget, top\_position)

Argument Information

| Argument     | Type              | Access | Mechanism |
|--------------|-------------------|--------|-----------|
| widget       | DXmCSTextWidget   | read   | value     |
| top_position | DXmCSTextPosition | read   | value     |

---

## DXmCSTextShowPosition

### FORMAT

DXmCSTextShowPosition  
(widget, position)

Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| widget   | DXmCSTextWidget   | read   | value     |
| position | DXmCSTextPosition | read   | value     |

---

## DXmCSTextVerticalScroll

### FORMAT

DXmCSTextVerticalScroll  
(widget, n)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| widget   | DXmCSTextWidget | read   | value     |
| n        | LONGWORD        | read   | value     |

---

## DXmCSTextXYToPos

### FORMAT

result DXmCSTextXYToPos  
(widget, x, y)

### Argument Information

| Argument | Type              | Access | Mechanism |
|----------|-------------------|--------|-----------|
| result   | DXmCSTextPosition | write  | value     |
| widget   | Widget            | read   | value     |
| x        | Position          | read   | value     |
| y        | Position          | read   | value     |

---

## DXmCvtCStoDDIF

### FORMAT

result DXmCvtCStoDDIF  
(cs, byte\_count, status)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Opaque | write  | value     |

## DECwindows Toolkit Extensions Routines DXmCvtCStoDDIF

---

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| cs         | XmString | read   | value     |
| byte_count | LONGWORD | write  | reference |
| status     | LONGWORD | write  | reference |

---

---

### DXmCvtCStoFC

#### FORMAT

result DXmCvtCStoFC  
(cs, byte\_count, status)

#### Argument Information

---

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | Opaque   | write  | value     |
| cs         | XmString | read   | reference |
| byte_count | LONGWORD | write  | reference |
| status     | LONGWORD | write  | reference |

---

---

### DXmCvtCStoOS

#### FORMAT

result DXmCvtCStoOS  
(cs, byte\_count, status)

#### Argument Information

---

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | Opaque   | write  | value     |
| cs         | XmString | read   | value     |
| byte_count | LONGWORD | write  | reference |
| status     | LONGWORD | write  | reference |

---

## DECwindows Toolkit Extensions Routines DXmCvtDDIFtoCS

---

### DXmCvtDDIFtoCS

#### FORMAT

result DXmCvtDDIFtoCS  
(ddif, byte\_count, status)

#### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | XmString | write  | value     |
| ddif       | Opaque   | read   | value     |
| byte_count | LONGWORD | write  | reference |
| status     | LONGWORD | write  | reference |

---

### DXmCvtFCtoCS

#### FORMAT

result DXmCvtFCtoCS  
(fc, byte\_count, status)

#### Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | XmString | write  | value     |
| fc         | Opaque   | read   | value     |
| byte_count | LONGWORD | write  | reference |
| status     | LONGWORD | write  | reference |

---

### DXmCvtOStoCS

#### FORMAT

result DXmCvtOStoCS  
(os\_string, byte\_count, status)



Argument Information

| Argument   | Type     | Access | Mechanism |
|------------|----------|--------|-----------|
| result     | XmString | write  | value     |
| os_string  | Opaque   | read   | value     |
| byte_count | LONGWORD | write  | reference |
| status     | LONGWORD | write  | reference |

---

## DXmDisplayCSMessage

### FORMAT

result DXmDisplayCSMessage  
(parent, name, pos, x, y, style, msgvec, widget, user\_routine, ok\_callback, help\_callback)

Argument Information

| Argument      | Type          | Access  | Mechanism |
|---------------|---------------|---------|-----------|
| result        | Widget        | write   | value     |
| parent        | Widget        | read    | value     |
| name          | XtString      | read    | reference |
| pos           | LONGWORD      | read    | value     |
| x             | LONGWORD      | read    | value     |
| y             | LONGWORD      | read    | value     |
| style         | LONGWORD      | read    | value     |
| msgvec        | ADDRESS       | read    | reference |
| widget        | Widget        | modifer | reference |
| user_routine  | ADDRESS       | read    | value     |
| ok_callback   | XtCallbackRec | read    | reference |
| help_callback | XtCallbackRec | read    | reference |

---

## DXmDisplayVmsMessage

### FORMAT

result DXmDisplayVmsMessage  
(parent, name, pos, x, y, style, msgvec, widget, user\_routine, ok\_callback, help\_callback )

## DECwindows Toolkit Extensions Routines DXmDisplayVmsMessage

### Argument Information

| Argument      | Type          | Access   | Mechanism |
|---------------|---------------|----------|-----------|
| result        | Widget        | write    | value     |
| parent        | Widget        | read     | value     |
| name          | XtString      | read     | reference |
| pos           | LONGWORD      | read     | value     |
| x             | LONGWORD      | read     | value     |
| y             | LONGWORD      | read     | value     |
| style         | LONGWORD      | read     | value     |
| msgvec        | ADDRESS       | read     | reference |
| widget        | Widget        | modifier | reference |
| user_routine  | ADDRESS       | read     | reference |
| ok_callback   | XtCallbackRec | read     | reference |
| help_callback | XtCallbackRec | read     | reference |

---

## DXmFindFontFallback

### FORMAT

result DXmFindFontFallback  
(fontname )

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XtString | write  | value     |
| fontname | XtString | read   | reference |

---

## DXmFontListCreateDefault

### FORMAT

result DXmFontListCreateDefault  
(widget)

**Argument Information**

| Argument | Type       | Access | Mechanism |
|----------|------------|--------|-----------|
| result   | XmFontList | write  | reference |
| widget   | Widget     | read   | value     |

---

**DXmFormSpaceButtonsEqually**

**FORMAT**

DXmFormSpaceButtonsEqually  
(parent, widget\_list, num\_widgets)

**Argument Information**

| Argument    | Type           | Access | Mechanism |
|-------------|----------------|--------|-----------|
| parent      | Widget         | read   | value     |
| widget_list | Widget (array) | read   | reference |
| num_widgets | Cardinal       | read   | value     |

---

**DXmGetLocaleCharset**

**FORMAT**

result DXmGetLocaleCharset  
( )

**Argument Information**

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | XmStringCharSet | write  | reference |

---

## DXmGetLocaleCharsets

### FORMAT

result DXmGetLocaleCharsets  
( )

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| result   | ADDRESS | write  | reference |

---

## DXmGetLocaleMnemonic

### FORMAT

result DXmGetLocaleMnemonic  
(context, widget, mnemonic, charset)

### Argument Information

| Argument | Type            | Access | Mechanism |
|----------|-----------------|--------|-----------|
| result   | KeySym          | write  | reference |
| context  | I18nContext     | read   | value     |
| widget   | Widget          | read   | value     |
| mnemonic | XmString        | read   | reference |
| charset  | XmStringCharSet | read   | value     |

---

## DXmGetLocaleString

### FORMAT

result DXmGetLocaleString  
(ascii)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | XmString | write  | value     |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| ascii    | XtString | read   | reference |

---

## DXmHelpOnContext

### FORMAT

DXmHelpOnContext  
(widget, confine)

### Argument Information

| Argument | Type    | Access | Mechanism |
|----------|---------|--------|-----------|
| widget   | Widget  | read   | value     |
| confine  | Boolean | read   | value     |

---

## DXmHelpSystemClose

### FORMAT

DXmHelpSystemClose  
(help\_context, routine, tag)

### Argument Information

| Argument     | Type    | Access | Mechanism |
|--------------|---------|--------|-----------|
| help_context | Opaque  | read   | value     |
| routine      | ADDRESS | read   | value     |
| tag          | Opaque  | read   | value     |

---

## DXmHelpSystemDisplay

### FORMAT

DXmHelpSystemDisplay  
(help\_context, help\_file, keyword, name, routine, tag)

## DECwindows Toolkit Extensions Routines

### DXmHelpSystemDisplay

#### Argument Information

---

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| help_context | Opaque   | read   | value     |
| help_file    | XtString | read   | reference |
| keyword      | XtString | read   | reference |
| name         | XtString | read   | reference |
| routine      | ADDRESS  | read   | value     |
| tag          | Opaque   | read   | value     |

---

---

## DXmHelpSystemOpen

### FORMAT

DXmHelpSystemOpen

(help\_context, main\_window, help\_file, routine, tag)

#### Argument Information

---

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| help_context | Opaque   | write  | reference |
| main_window  | Widget   | read   | value     |
| help_file    | XtString | read   | reference |
| routine      | ADDRESS  | read   | value     |
| tag          | Opaque   | read   | value     |

---

---

## DXmInitialize

### FORMAT

DXmInitialize

()

---

## DXmLoadQueryFont

### FORMAT

result DXmLoadQueryFont

(dpy, fontname)

**Argument Information**

| Argument | Type                 | Access | Mechanism |
|----------|----------------------|--------|-----------|
| result   | LONGWORD<br>UNSIGNED | write  | value     |
| dpy      | Display              | read   | value     |
| fontname | XtString             | read   | reference |

**DXmNumChildren**

**FORMAT**

result DXmNumChildren  
(widget)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | Cardinal | write  | value     |
| widget   | Widget   | read   | value     |

**DXmPositionWidget**

**FORMAT**

DXmPositionWidget  
(new\_widget, avoid\_widgets, widget\_cnt)

**Argument Information**

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| new_widget    | Widget   | read   | value     |
| avoid_widgets | Widget   | modify | reference |
| widget_cnt    | LONGWORD | read   | value     |

## DECwindows Toolkit Extensions Routines DXmPrintWgtAugmentList

---

### DXmPrintWgtAugmentList

#### FORMAT

result DXmPrintWgtAugmentList  
(ar\_pw, 1\_list, ar\_data)

#### Argument Information

| Argument | Type                 | Access | Mechanism |
|----------|----------------------|--------|-----------|
| result   | LONGWORD<br>UNSIGNED | write  | value     |
| ar_pw    | Widget               | read   | value     |
| 1_list   | LONGWORD             | read   | value     |
| ar_data  | XtPointer            | read   | value     |

---

### DXmPrintWgtPrintJob

#### FORMAT

result DXmPrintWgtPrintJob  
(ar\_pw, ar\_cs\_filenames, 1\_filename\_count)

#### Argument Information

| Argument         | Type                 | Access | Mechanism |
|------------------|----------------------|--------|-----------|
| result           | LONGWORD<br>UNSIGNED | write  | value     |
| ar_pw            | Widget               | read   | value     |
| ar_cs_filenames  | XmString             | read   | reference |
| 1_filename_count | LONGWORD             | read   | value     |

---

### DXmStringCheck

#### FORMAT

result DXmStringCheck  
(old\_external, new\_external)



**Argument Information**

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| result       | Boolean  | write  | value     |
| old_external | XmString | write  | reference |
| new_external | XmString | write  | reference |

---

**DXmSvnAddEntries**

**FORMAT**

DXmSvnAddEntries  
(svnw, after\_entry, number\_of\_entries, level, entry\_tags, index\_window)

**Argument Information**

| Argument          | Type                 | Access | Mechanism |
|-------------------|----------------------|--------|-----------|
| svnw              | Widget               | read   | value     |
| after_entry       | LONGWORD             | read   | value     |
| number_of_entries | LONGWORD             | read   | value     |
| level             | LONGWORD             | read   | value     |
| entry_tags        | LONGWORD<br>UNSIGNED | modify | reference |
| index_window      | LONGWORD             | read   | value     |

---

**DXmSvnAutoScrollCheck**

**FORMAT**

result DXmSvnAutoScrollCheck  
(svnw, x, y)

**Argument Information**

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| svnw     | Widget   | read   | value     |
| x        | LONGWORD | read   | value     |
| y        | LONGWORD | read   | value     |

---

## DXmSvnAutoScrollDisplay

### FORMAT

DXmSvnAutoScrollDisplay  
(svnw, x, y)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| svnw     | Widget   | read   | value     |
| x        | LONGWORD | read   | value     |
| y        | LONGWORD | read   | value     |

---

## DXmSvnClearHighlight

### FORMAT

DXmSvnClearHighlight  
(svnw, entry\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnClearHighlighting

### FORMAT

DXmSvnClearHighlighting  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnClearSelection

### FORMAT

DXmSvnClearSelection  
(svnw, entry\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnClearSelections

### FORMAT

DXmSvnClearSelections  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnDeleteEntries

### FORMAT

DXmSvnDeleteEntries  
(svnw, after\_entry, number\_of\_entries)

### Argument Information

| Argument          | Type     | Access | Mechanism |
|-------------------|----------|--------|-----------|
| svnw              | Widget   | read   | value     |
| after_entry       | LONGWORD | read   | value     |
| number_of_entries | LONGWORD | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmSvnDisableDisplay

---

## DXmSvnDisableDisplay

### FORMAT

DXmSvnDisableDisplay  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnEnableDisplay

### FORMAT

DXmSvnEnableDisplay  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnFlushEntry

### FORMAT

DXmSvnFlushEntry  
(svnw, entry\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnGetComponentNumber

### FORMAT

result DXmSvnGetComponentNumber  
(svnw, comp\_tag)

### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| result   | LONGWORD  | write  | value     |
| svnw     | Widget    | read   | value     |
| comp_tag | XtPointer | read   | value     |

---

## DXmSvnGetComponentTag

### FORMAT

result DXmSvnGetComponentTag  
(svnw, comp\_number)

### Argument Information

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| result      | XtPointer | write  | value     |
| svnw        | Widget    | read   | value     |
| comp_number | LONGWORD  | read   | value     |

---

## DXmSvnGetComponentWidth

### FORMAT

result DXmSvnGetComponentWidth  
(svnw, comp\_number)

### Argument Information

| Argument | Type      | Access | Mechanism |
|----------|-----------|--------|-----------|
| result   | Dimension | write  | value     |

## DECwindows Toolkit Extensions Routines

### DXmSvnGetComponentWidth

| Argument    | Type     | Access | Mechanism |
|-------------|----------|--------|-----------|
| svnw        | Widget   | read   | value     |
| comp_number | LONGWORD | read   | value     |

---

## DXmSvnGetComponentText

### FORMAT

result DXmSvnGetComponentText  
(svnw, entry\_number, comp\_number, result )

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| result       | XmString | write  | value     |
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| comp_number  | LONGWORD | read   | value     |

---

## DXmSvnGetDisplayed

### FORMAT

DXmSvnGetDisplayed  
(svnw, entry\_number, entry\_tags, Y\_coords, num\_array\_entries)

### Argument Information

| Argument          | Type     | Access | Mechanism |
|-------------------|----------|--------|-----------|
| svnw              | Widget   | read   | value     |
| entry_number      | LONGWORD | modify | reference |
| entry_tags        | LONGWORD | modify | reference |
| y_coords          | LONGWORD | modify | reference |
| num_array_entries | LONGWORD | read   | value     |

---

## DXmSvnGetEntryLevel

### FORMAT

result DXmSvnGetEntryLevel  
(svnw, entry\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| result       | LONGWORD | write  | value     |
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnGetEntryNumber

### FORMAT

result DXmSvnGetEntryNumber  
(svnw, tag)

### Argument Information

| Argument | Type                 | Access | Mechanism |
|----------|----------------------|--------|-----------|
| result   | LONGWORD<br>UNSIGNED | write  | value     |
| svnw     | Widget               | read   | value     |
| tag      | XtPointer            | read   | value     |

---

## DXmSvnGetEntryPosition

### FORMAT

DXmSvnGetEntryPosition  
(svnw, entry\_number, window\_mode, x, y)

## DECwindows Toolkit Extensions Routines

### DXmSvnGetEntryPosition

#### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| window_mode  | Boolean  | read   | value     |
| x            | LONGWORD | read   | value     |
| y            | LONGWORD | read   | value     |

---

## DXmSvnGetEntrySensitivity

#### FORMAT

result DXmSvnGetEntrySensitivity  
(svnw, entry\_number, result)

#### Argument Information

| Argument     | Type                 | Access | Mechanism |
|--------------|----------------------|--------|-----------|
| result       | LONGWORD<br>UNSIGNED | write  | value     |
| svnw         | Widget               | read   | value     |
| entry_number | LONGWORD             | read   | value     |
| window_mode  | Boolean              | read   | value     |

---

## DXmSvnGetEntryTag

#### FORMAT

result DXmSvnGetEntryTag  
(svnw, entry\_number)

#### Argument Information

| Argument     | Type      | Access | Mechanism |
|--------------|-----------|--------|-----------|
| result       | XtPointer | write  | value     |
| svnw         | Widget    | read   | value     |
| entry_number | LONGWORD  | read   | value     |



---

## DXmSvnGetHighlighted

### FORMAT

DXmSvnGetHighlighted  
(svnw, highlighted, entry\_tags, num\_array\_entries)

### Argument Information

| Argument          | Type     | Access | Mechanism |
|-------------------|----------|--------|-----------|
| svnw              | Widget   | read   | value     |
| highlighted       | LONGWORD | modify | reference |
| entry_tags        | LONGWORD | modify | reference |
| num_array_entries | LONGWORD | read   | value     |

---

## DXmSvnGetNumDisplayed

### FORMAT

result DXmSvnGetNumDisplayed  
(svnw)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |
| svnw     | Widget   | read   | value     |

---

## DXmSvnGetNumHighlighted

### FORMAT

result DXmSvnGetNumHighlighted  
(svnw)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| result   | LONGWORD | write  | value     |

## DECwindows Toolkit Extensions Routines

### DXmSvnGetNumHighlighted

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnGetNumSelections

### FORMAT

result DXmSvnGetNumSelections  
(svnw, selections, comps, entry\_tags, num\_array\_entries)

### Argument Information

| Argument          | Type                 | Access | Mechanism |
|-------------------|----------------------|--------|-----------|
| result            | LONGWORD             | write  | value     |
| svnw              | Widget               | read   | value     |
| selections        | LONGWORD             | modify | reference |
| comps             | LONGWORD             | modify | reference |
| entry_tags        | LONGWORD<br>UNSIGNED | modify | reference |
| num_array_entries | LONGWORD             | read   | value     |

---

## DXmSvnGetPrimaryWorkWidget

### FORMAT

result DXmSvnGetPrimaryWorkWidget  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |
| svnw     | Widget | read   | value     |

---

## DXmSvnGetSecondaryWorkWidget

### FORMAT

result DXmSvnGetSecondaryWorkWidget  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| result   | Widget | write  | value     |
| svnw     | Widget | read   | value     |

---

## DXmSvnGetSelections

### FORMAT

DXmSvnGetSelections  
(svnw, selections, comps, entry\_tags, num\_array\_entries)

### Argument Information

| Argument          | Type                 | Access | Mechanism |
|-------------------|----------------------|--------|-----------|
| svnw              | Widget               | read   | value     |
| selections        | LONGWORD             | modify | reference |
| comps             | LONGWORD             | modify | reference |
| entry_tags        | LONGWORD<br>UNSIGNED | modify | reference |
| num_array_entries | LONGWORD             | read   | value     |

---

## DXmSvnGetTreePosition

### FORMAT

DXmSvnGetTreePosition  
(svnw, x, y)

## DECwindows Toolkit Extensions Routines

### DXmSvnGetTreePosition

#### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| svnw     | Widget   | read   | value     |
| x        | LONGWORD | write  | reference |
| y        | LONGWORD | write  | reference |

---

## DXmSvnHideHighlighting

### FORMAT

DXmSvnHideHighlighting  
(svnw)

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnHideSelections

### FORMAT

DXmSvnHideSelections  
(svnw)

#### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnHighlightAll

### FORMAT

DXmSvnHighlightAll  
(svnw)

Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnHighlightEntry

FORMAT

DXmSvnHighlightEntryRoutine  
(svnw, entry\_number)

Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnInsertComponent

FORMAT

DXmSvnInsertComponent  
(svnw, comp\_number, width, tag)

Argument Information

| Argument    | Type                 | Access | Mechanism |
|-------------|----------------------|--------|-----------|
| svnw        | Widget               | read   | value     |
| comp_number | LONGWORD             | read   | value     |
| width       | Dimension            | read   | value     |
| tag         | LONGWORD<br>UNSIGNED | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmSvnInvalidateEntry

---

## DXmSvnInvalidateEntry

### FORMAT

DXmSvnInvalidateEntry  
(svnw, entry\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnMapPosition

### FORMAT

DXmSvnMapPosition  
(svnw, find\_x, find\_y, entry\_number, component, tag)

### Argument Information

| Argument     | Type                 | Access | Mechanism |
|--------------|----------------------|--------|-----------|
| svnw         | Widget               | read   | value     |
| find_x       | LONGWORD             | read   | value     |
| find_y       | LONGWORD             | read   | value     |
| entry_number | LONGWORD             | modify | reference |
| component    | LONGWORD             | modify | reference |
| tag          | LONGWORD<br>UNSIGNED | modify | reference |

---

## DXmSvnPosition Display

### FORMAT

result DXmSvnPositionDisplay  
(svnw, entry\_number, position)

**Argument Information**

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| result       | LONGWORD | write  | value     |
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| position     | LONGWORD | read   | value     |

---

**DXmSvnRemoveComponent**

**FORMAT**

DXmSvnRemoveComponent  
(svnw, comp\_number)

**Argument Information**

| Argument    | Type     | Access | Mechanism |
|-------------|----------|--------|-----------|
| svnw        | Widget   | read   | value     |
| comp_number | LONGWORD | read   | value     |

---

**DXmSvnSelectAll**

**FORMAT**

DXmSvnSelectAll  
(svnw)

**Argument Information**

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmSvnSelectComponent

---

## DXmSvnSelectComponent

### FORMAT

DXmSvnSelectComponent  
(svnw, entry\_number, comp\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| comp_number  | LONGWORD | read   | value     |

---

## DXmSvnSelectEntry

### FORMAT

DXmSvnSelectEntry  
(svnw, entry\_number)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |

---

## DXmSvnSetAppIDragging

### FORMAT

DXmSvnSetAppIDragging  
(svnw, value)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| svnw     | Widget   | read   | value     |
| value    | LONGWORD | read   | value     |



---

## DXmSvnSetComponentHidden

### FORMAT

DXmSvnSetComponentHidden  
(svnw, entry\_number, comp\_number, hidden\_mode)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| comp_number  | LONGWORD | read   | value     |
| hidden_mode  | LONGWORD | read   | value     |

---

## DXmSvnSetComponentPixmap

### FORMAT

DXmSvnSetComponentPixmap  
(svnw, entry\_number, comp\_number, x, y, pixmap, width, height)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| comp_number  | LONGWORD | read   | value     |
| x            | LONGWORD | read   | value     |
| y            | LONGWORD | read   | value     |
| pixmap       | Pixmap   | read   | value     |
| width        | LONGWORD | read   | value     |
| height       | LONGWORD | read   | value     |

## DECwindows Toolkit Extensions Routines

### DXmSvnSetComponentTag

---

## DXmSvnSetComponentTag

### FORMAT

DXmSvnSetComponentTag  
(svnw, comp\_number, tag)

### Argument Information

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| svnw        | Widget    | read   | value     |
| comp_number | LONGWORD  | read   | value     |
| tag         | XtPointer | write  | value     |

---

## DXmSvnSetComponentText

### FORMAT

DXmSvnSetComponentText  
(svnw, entry\_number, comp\_number, x, y, cs, fontlist)

### Argument Information

| Argument     | Type                 | Access | Mechanism |
|--------------|----------------------|--------|-----------|
| svnw         | Widget               | read   | value     |
| entry_number | LONGWORD             | read   | value     |
| comp_number  | LONGWORD             | read   | value     |
| x            | LONGWORD             | read   | value     |
| y            | LONGWORD             | read   | value     |
| cs           | LONGWORD<br>UNSIGNED | read   | value     |
| fontlist     | LONGWORD<br>UNSIGNED | read   | value     |

---

## DXmSvnSetComponentWidget

### FORMAT

DXmSvnSetComponentWidget  
(svnw, entry\_number, comp\_number, x, y, subw)

### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| comp_number  | LONGWORD | read   | value     |
| x            | LONGWORD | read   | value     |
| y            | LONGWORD | read   | value     |
| subw         | Widget   | read   | value     |

---

## DXmSvnSetComponentWidth

### FORMAT

DXmSvnSetComponentWidth  
(svnw, comp\_number, width)

### Argument Information

| Argument    | Type      | Access | Mechanism |
|-------------|-----------|--------|-----------|
| svnw        | Widget    | read   | value     |
| comp_number | LONGWORD  | read   | value     |
| width       | Dimension | read   | value     |

---

## DXmSvnSetEntry

### FORMAT

DXmSvnSetEntry  
(svnw, entry\_number, width, height, num\_components, sensitivity, entry\_tag, index\_window)

## DECwindows Toolkit Extensions Routines

### DXmSvnSetEntry

#### Argument Information

| Argument       | Type                 | Access | Mechanism |
|----------------|----------------------|--------|-----------|
| svnw           | Widget               | read   | value     |
| entry_number   | LONGWORD             | read   | value     |
| width          | LONGWORD             | read   | value     |
| height         | LONGWORD             | read   | value     |
| num_components | LONGWORD             | read   | value     |
| sensitivity    | Boolean              | read   | value     |
| entry_tag      | LONGWORD<br>UNSIGNED | read   | value     |
| index_window   | Boolean              | read   | value     |

---

## DXmSvnSetEntryIndexWindow

### FORMAT

DXmSvnSetEntryIndexWindow  
(svnw, entry\_number, index\_window)

#### Argument Information

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| index_window | Boolean  | read   | value     |

---

## DXmSvnSetEntryNumComponents

### FORMAT

DXmSvnSetEntryNumComponents  
(svnw, entry\_numbers, num\_components)

#### Argument Information

| Argument      | Type     | Access | Mechanism |
|---------------|----------|--------|-----------|
| svnw          | Widget   | read   | value     |
| entry_numbers | LONGWORD | read   | value     |

## DECwindows Toolkit Extensions Routines DXmSvnSetEntryNumComponents

---

| Argument       | Type     | Access | Mechanism |
|----------------|----------|--------|-----------|
| num_components | LONGWORD | read   | value     |

---

---

### DXmSvnSetEntryPosition

#### FORMAT

DXmSvnSetEntryPosition  
(svnw, entry\_number, window\_mode, x, y)

#### Argument Information

---

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| window_mode  | Boolean  | read   | value     |
| x            | LONGWORD | write  | reference |
| y            | LONGWORD | write  | reference |

---

---

### DXmSvnSetEntrySensitivity

#### FORMAT

DXmSvnSetEntrySensitivity  
(svnw, entry\_number, sensitivity)

#### Argument Information

---

| Argument     | Type     | Access | Mechanism |
|--------------|----------|--------|-----------|
| svnw         | Widget   | read   | value     |
| entry_number | LONGWORD | read   | value     |
| sensitivity  | Boolean  | read   | value     |

---

---

## DXmSvnSetEntryTag

### FORMAT

DXmSvnSetEntryTag  
(svnw, entry\_number, entry\_tag)

### Argument Information

| Argument     | Type                 | Access | Mechanism |
|--------------|----------------------|--------|-----------|
| svnw         | Widget               | read   | value     |
| entry_number | LONGWORD             | read   | value     |
| entry_tag    | LONGWORD<br>UNSIGNED | read   | value     |

---

## DXmSvnShowHighlighting

### FORMAT

DXmSvnShowHighlighting  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnSetTreePosition

### FORMAT

DXmSvnSetTreePosition  
(svnw, x, y)

### Argument Information

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| svnw     | Widget   | read   | value     |
| x        | LONGWORD | read   | value     |

| Argument | Type     | Access | Mechanism |
|----------|----------|--------|-----------|
| y        | LONGWORD | read   | value     |

---

## DXmSvnShowSelections

### FORMAT

DXmSvnShowSelections  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnValidateAll

### FORMAT

DXmSvnValidateAll  
(svnw)

### Argument Information

| Argument | Type   | Access | Mechanism |
|----------|--------|--------|-----------|
| svnw     | Widget | read   | value     |

---

## DXmSvnWidget

### FORMAT

result DXmSvnWidget  
(parent, name, x, y, width, height, confirm\_callback, help\_callback,  
attach\_callback, detach\_callback, get\_entry\_callback, extend\_confirm\_callback,  
entry\_selected\_callback, entry\_unselected\_callback, selections\_dragged\_callback,  
dragging\_callback, dragging\_end\_callback)

## DECwindows Toolkit Extensions Routines DXmSvnWidget

### Argument Information

| Argument                    | Type           | Access | Mechanism |
|-----------------------------|----------------|--------|-----------|
| result                      | Widget         | write  | value     |
| parent                      | Widget         | read   | value     |
| name                        | XtString       | read   | reference |
| x                           | LONGWORD       | read   | value     |
| y                           | LONGWORD       | read   | value     |
| height                      | LONGWORD       | read   | value     |
| width                       | LONGWORD       | read   | value     |
| confirm_callback            | XtCallbackList | read   | reference |
| help_callback               | XtCallbackList | read   | reference |
| attach_callback             | XtCallbackList | read   | reference |
| detach_callback             | XtCallbackList | read   | reference |
| get_entry_callback          | XtCallbackList | read   | reference |
| extend_confirm_callback     | XtCallbackList | read   | reference |
| entry_selected_callback     | XtCallbackList | read   | reference |
| entry_unselected_callback   | XtCallbackList | read   | reference |
| selections_dragged_callback | XtCallbackList | read   | reference |
| dragging_callback           | XtCallbackList | read   | reference |
| dragging_end_callback       | XtCallbackList | read   | reference |



---

## DECwindows Toolkit Extensions Data Structures

This chapter documents DECwindows toolkit extensions data structures.

---

### DXmColorMixCallbackStruct

#### Field Information

| Argument | Type          |
|----------|---------------|
| reason   | LONGWORD      |
| event    | address       |
| newred   | WORD UNSIGNED |
| newgrn   | WORD UNSIGNED |
| newblu   | WORD UNSIGNED |
| newname  | XtString      |
| origred  | WORD UNSIGNED |
| origgrn  | WORD UNSIGNED |
| origblu  | WORD UNSIGNED |

---

### DXmCSTextCallbackStruct

#### Field Information

| Argument    | Type     |
|-------------|----------|
| reason      | LONGWORD |
| event       | ADDRESS  |
| charset     | ADDRESS  |
| charset_len | LONGWORD |

---

---

## DXmCSTextVerifyCallbackStruct

### Field Information

---

| Argument   | Type     |
|------------|----------|
| reason     | LONGWORD |
| event      | ADDRESS  |
| doit       | Boolean  |
| currInsert | LONGWORD |
| newInsert  | LONGWORD |
| startPos   | LONGWORD |
| endPos     | LONGWORD |
| text       | ADDRESS  |

---

---

## DXmPrintFormatStruct

### Field Information

---

| Argument   | Type     |
|------------|----------|
| ui_string  | XmString |
| os_string  | XmString |
| var_string | XmString |

---

---

## DxmPrintOptionsMenuStruct

### Field Information

---

| Argument  | Type     |
|-----------|----------|
| ui_string | XmString |
| os_string | XmString |

---

---

## DXmSvnCallbackStruct

### Field Information

---

| Argument                | Type      |
|-------------------------|-----------|
| reason                  | LONGWORD  |
| entry_number            | LONGWORD  |
| component_number        | LONGWORD  |
| first_selection         | LONGWORD  |
| x                       | LONGWORD  |
| y                       | LONGWORD  |
| entry_tag               | XtPointer |
| time                    | LONGWORD  |
| entry_level             | LONGWORD  |
| loc_cursor_entry_number | LONGWORD  |
| transfer_mode           | LONGWORD  |
| dragged_entry_number    | LONGWORD  |
| event                   | ADDRESS   |

---



---

# Index

## A

---

Access control  
  disabling, 2-38, 2-174  
  enabling, 2-54, 2-174  
Access type, 1-4, 1-10  
ACTIVATE SCREEN SAVER routine, 2-1  
ADD HOST routine, 2-1  
ADD HOSTS routine, 2-2  
Adding  
  resource, 2-162, 2-165  
  resource entry, 2-161  
ADD PIXEL routine, 2-2  
ADD TO SAVE SET routine, 2-2  
  See also REMOVE FROM SAVE SET routine  
Allocating quark, 2-168  
Allocating storage, 2-134  
Allocation of color planes, 2-5  
ALLOC CLASS HINT routine, 2-3  
ALLOC COLOR CELLS routine, 2-4  
ALLOC COLOR PLANES routine, 2-5  
ALLOC COLOR routine, 2-3  
ALLOC ICON SIZE routine, 2-5  
ALLOC NAMED COLOR routine, 2-6  
ALLOC SIZE HINTS routine, 2-6  
ALLOC STANDARD COLORMAP routine, 2-7  
ALLOW EVENTS routine, 2-7  
ALL PLANES routine, 2-8  
Any event data structure, 3-1  
Arc  
  drawing, 2-45  
  drawing multiple, 2-46  
  filling, 2-58  
  filling multiple, 2-58  
Arc data structure, 3-2  
Arc mode  
  setting, 2-175  
Arg data structure, 5-1  
Atom  
  returning identifier, 2-110  
  returning name, 2-68  
Auto-repeat  
  changing value, 2-13  
  obtaining value, 2-80

AUTO REPEAT OFF routine, 2-8  
AUTO REPEAT ON routine, 2-8

## B

---

Background  
  changing in graphics context, 2-176, 2-193  
  window  
    See Window background  
Background tile  
  window  
    See Window background tile  
Backing store, 2-44  
Bell, 2-13  
  obtaining value, 2-80  
BELL routine, 2-9  
Bitmap  
  creating from data, 2-25  
  leftmost bit, 2-9  
  reading file, 2-147  
  unit, 2-10  
  writing file, 2-223  
BITMAP BIT ORDER routine, 2-9  
BITMAP PAD routine, 2-10  
BITMAP UNIT routine, 2-10  
Bit plane  
  copying, 2-24  
BLACK PIXEL OF SCREEN routine, 2-11  
BLACK PIXEL routine, 2-10  
Border  
  See Window border  
Button  
  passive grabbing, 2-100  
  ungrabbing, 2-214  
Button event data structure, 3-3  
By descriptor passing mechanism, 1-5  
By reference passing mechanism, 1-5, 1-11  
By value passing mechanism, 1-5, 1-11

## C

---

Cap style  
  changing, 2-186  
CHANGE ACTIVE POINTER GRAB routine,  
  2-11

- CHANGE GC routine, 2-12
- CHANGE KEYBOARD CONTROL routine, 2-13
- CHANGE KEYBOARD MAPPING routine, 2-13
- CHANGE POINTER CONTROL routine, 2-14
- CHANGE PROPERTY routine, 2-14
- CHANGE SAVE SET routine, 2-15
- CHANGE WINDOW ATTRIBUTES routine, 2-16
- Changing cap style, 2-186
- Changing clip mask, 2-177
- Changing color definition, 2-204, 2-205
  - by name, 2-206
- Changing corner patterns, 2-186
- Changing endpoint patterns, 2-186
- Changing join style, 2-186
- Changing line style, 2-186
- Changing line width, 2-186
- Changing window attributes, 2-16
- Char 2B data structure, 3-4
- Char struct data structure, 3-5
- CHECK IF EVENT routine, 2-16
  - See also IF EVENT routine
- CHECK MASK EVENT routine, 2-16
  - See also MASK EVENT routine
- CHECK TYPED EVENT routine, 2-17
- CHECK TYPED WINDOW EVENT routine, 2-17
- CHECK WINDOW EVENT routine, 2-18
  - See also WINDOW EVENT routine
- Circulate event data structure, 3-5
- Circulate request event data structure, 3-6
- CIRCULATE SUBWINDOWS DOWN routine, 2-19
- CIRCULATE SUBWINDOWS routine, 2-18
- CIRCULATE SUBWINDOWS UP routine, 2-19
- Circulating subwindow, 2-18, 2-19
- Class hint data structure, 3-7
- CLEAR AREA routine, 2-19
- Clearing a rectangular area, 2-19
- Clearing a window, 2-20
- CLEAR WINDOW routine, 2-20
- Client
  - kill, 2-112
  - sending event to, 2-173
- Client message event data structure, 3-7
- CLIP BOX routine, 2-20
- Clip origin
  - setting in graphics context, 2-177
- Clipping
  - changing clip mask, 2-177
  - setting clip rectangles, 2-178
  - setting origin in graphics context, 2-177
- CLOSE DISPLAY routine, 2-21
- Closing display, 2-21
- Closing down
  - defining client resources, 2-178
  - forced, 2-112
- Color data structure, 3-8
- Color definition
  - changing, 2-204, 2-205
    - by name, 2-206
  - closest color, 2-3
  - deallocating, 2-64
  - determining value, 2-141, 2-142
  - obtaining, 2-122
  - obtaining values, 2-131
  - reserving, 2-4
  - setting, 2-204
  - specifying, 2-3
- Color index
  - black, 2-10
  - white, 2-220
- Color map
  - cells, 2-11, 2-39
  - creating, 2-23, 2-25
  - default, 2-31
  - freeing, 2-63
  - installing, 2-109
  - installing new, 2-216
  - list of installed, 2-118
  - obtaining maximum supported by screen, 2-126
  - obtaining minimum supported by screen, 2-127
  - replacing, 2-198
  - standard
    - obtaining, 2-88
    - setting, 2-192
- Color map event data structure, 3-9
- Color plane
  - allocating, 2-5
- Color Routines
  - ALLOC COLOR, 2-3
  - ALLOC COLOR CELLS, 2-4
  - ALLOC COLOR PLANES, 2-5
  - ALLOC NAMED COLOR, 2-6
  - ALLOC STANDARD COLORMAP, 2-7
  - COPY COLORMAP AND FREE, 2-23
  - CREATE COLORMAP, 2-25
  - FREE COLORMAP, 2-63
  - FREE COLORS, 2-64
  - GET RGB COLORMAPS, 2-86
  - GET STANDARD COLORMAP, 2-88
  - GET VISUAL INFO, 2-91
  - GET WM COLORMAP WINDOWS, 2-95
  - LOOKUP COLOR, 2-122
  - MATCH VISUAL INFO, 2-125
  - QUERY COLOR, 2-141
  - QUERY COLORS, 2-142
  - SET RGB COLORMAPS, 2-190
  - SET STANDARD COLORMAP, 2-192
  - SET WINDOW COLORMAP, 2-198
  - SET WM COLORMAP WINDOWS, 2-199
  - STORE COLOR, 2-204
  - STORE COLORS, 2-205

## Color Routines (cont'd)

- STORE NAMED COLOR, 2-206
- Color value, 2-131
  - allocating, 2-6
- Compose data structure, 3-10
- Compose status data structure, 3-10
- Configure event data structure, 3-11
- Configure request event data structure, 3-12
- CONFIGURE WINDOW routine, 2-21
- Configuring a window, 2-21
- Connection number
  - returning, 2-22
- CONNECTION NUMBER routine, 2-22
- Converting
  - quark to string, 2-166
  - string to binding list, 2-167
  - string to quark, 2-167
  - string to quark list, 2-167
- CONVERT SELECTION routine, 2-22
- Coordinates
  - transferring, 2-212
- COPY AREA routine, 2-23
- COPY COLORMAP AND FREE routine, 2-23
- COPY GC routine, 2-24
- Copying a bit plane, 2-24
- Copying graphics context, 2-24
- COPY PLANE routine, 2-24
- Corner patterns
  - changing, 2-186
- CREATE BITMAP FROM DATA routine, 2-25
- CREATE COLORMAP routine, 2-25
- CREATE FONT CURSOR routine, 2-26
- CREATE GC routine, 2-26
- CREATE GLYPH CURSOR routine, 2-27
- CREATE IMAGE routine, 2-27
- CREATE PIXMAP FROM BITMAP DATA routine, 2-29
- CREATE PIXMAP routine, 2-28
- CREATE REGION routine, 2-29
- CREATE SIMPLE WINDOW routine, 2-29
- Create window event data structure, 3-13
- CREATE WINDOW routine, 2-30
- Creating
  - database, 2-158
- Creating a cursor
  - from font glyphs, 2-27
  - from fonts, 2-26
- Creating a window, 2-29, 2-30
- Crossing event data structure, 3-15
- Cursor
  - best size, 2-139
  - changing the color, 2-149
  - defining for a window, 2-35
  - deleting, 2-64
  - determining size supported by hardware, 2-138
  - dissociating from a window, 2-213

## Cursor Routines

- CREATE FONT CURSOR, 2-26
  - CREATE GLYPH CURSOR, 2-27
  - DEFINE CURSOR, 2-35
  - FREE CURSOR, 2-64
  - QUERY BEST CURSOR, 2-138
  - RECOLOR CURSOR, 2-149
  - UNDEFINE CURSOR, 2-213
- Cut buffer
- obtaining data stored in, 2-56
  - rotating, 2-169
  - storing data in, 2-203
- Cut buffer zero, 2-56
- storing data in, 2-204

## D

---

### Dashed line

- changing length, 2-179
- changing offset, 2-179

### Database

- creating, 2-158
- loading, 2-160
- merging, 2-159
- retrieving, 2-157
- storing, 2-161

### Data structure

- description, 3-1
- visual information, 2-91

### Data type, 1-3

- Deallocating color definitions, 2-64
- DEFAULT COLORMAP OF SCREEN routine, 2-31
- DEFAULT COLORMAP routine, 2-31
- DEFAULT DEPTH routine, 2-31
- DEFAULT GC routine, 2-32
- DEFAULT ROOT WINDOW routine, 2-33
- DEFAULT SCREEN OF DISPLAY routine, 2-34
- DEFAULT SCREEN routine, 2-33
- DEFAULT VISUAL OF SCREEN routine, 2-35
- DEFAULT VISUAL routine, 2-34
- DEFINE CURSOR routine, 2-35
- DELETE CONTEXT routine, 2-35
- DELETE MODIFIERMAP ENTRY routine, 2-36
- DELETE PROPERTY routine, 2-37
- Deleting property, 2-37
- Descriptor passing mechanism, 1-5
- DESTROY IMAGE routine, 2-37
- Destroying all subwindows, 2-38
- Destroying a window, 2-38
- DESTROY REGION routine, 2-37
- DESTROY SUBWINDOWS routine, 2-38
- Destroy window event data structure, 3-17
- DESTROY WINDOW routine, 2-38
- DISABLE ACCESS CONTROL routine, 2-38
- Disconnection
  - See Closing down

Display  
   closing, 2-21  
   obtaining for specified screen, 2-42  
   opening, 2-130  
 DISPLAY CELLS routine, 2-39  
 DISPLAY HEIGHT MM routine, 2-40  
 DISPLAY HEIGHT routine, 2-39  
 DISPLAY KEYCODES routine, 2-40  
 DISPLAY MOTION BUFFER SIZE routine, 2-40  
 DISPLAY NAME routine, 2-41  
 DISPLAY OF SCREEN routine, 2-42  
 DISPLAY PLANES routine, 2-42  
 Display Routines  
   ALL PLANES, 2-8  
   BITMAP BIT ORDER, 2-9  
   BITMAP PAD, 2-10  
   BITMAP UNIT, 2-10  
   BLACK PIXEL, 2-10  
   BLACK PIXEL OF SCREEN, 2-11  
   CELLS OF SCREEN, 2-11  
   CLOSE DISPLAY, 2-21  
   CONNECTION NUMBER, 2-22  
   DEFAULT COLORMAP, 2-31  
   DEFAULT COLORMAP OF SCREEN, 2-31  
   DEFAULT DEPTH, 2-31  
   DEFAULT DEPTH OF SCREEN, 2-32  
   DEFAULT GC OF SCREEN, 2-32  
   DEFAULT ROOT WINDOW, 2-33  
   DEFAULT SCREEN, 2-33  
   DEFAULT SCREEN OF DISPLAY, 2-34  
   DEFAULT VISUAL, 2-34  
   DEFAULT VISUAL OF SCREEN, 2-35  
   DISPLAY CELLS, 2-39  
   DISPLAY HEIGHT, 2-39  
   DISPLAY HEIGHT MM, 2-40  
   DISPLAY NAME, 2-41  
   DISPLAY OF SCREEN, 2-42  
   DISPLAY PLANES, 2-42  
   DISPLAY STRING, 2-42  
   DISPLAY WIDTH, 2-43  
   DISPLAY WIDTH MM, 2-44  
   DOES BACKING STORE, 2-44  
   DOES SAVE UNDERS, 2-45  
   EVENT MASK OF SCREEN, 2-55  
   FREE, 2-63  
   HEIGHT MM OF SCREEN, 2-106  
   HEIGHT OF SCREEN, 2-106  
   IMAGE BYTE ORDER, 2-108  
   LAST KNOWN REQUEST PROCESSED,  
     2-112  
   LIST DEPTHS, 2-113  
   MAX CMAPS OF SCREEN, 2-126  
   MAX REQUEST SIZE, 2-126  
   MIN CMAPS OF SCREEN, 2-127  
   NEXT REQUEST, 2-129  
   NO OP, 2-129  
   OPEN DISPLAY, 2-130  
   PLANES OF SCREEN, 2-134

Display Routines (cont'd)  
   PROTOCOL REVISION, 2-136  
   PROTOCOL VERSION, 2-136  
   Q LENGTH, 2-138  
   ROOT WINDOW, 2-168  
   ROOT WINDOW OF SCREEN, 2-169  
   ROTATE BUFFERS, 2-169  
   SCREEN COUNT, 2-170  
   SCREEN NUMBER OF SCREEN, 2-171  
   SCREEN OF DISPLAY, 2-171  
   SCREEN RESOURCE STRING, 2-171  
   SERVER VENDOR, 2-174  
   STORE BUFFER, 2-203  
   STORE BYTES, 2-204  
   VENDOR RELEASE, 2-219  
   WHITE PIXEL, 2-220  
   WHITE PIXEL OF SCREEN, 2-221  
   WIDTH MM OF SCREEN, 2-221  
   WIDTH OF SCREEN, 2-221  
 DISPLAY STRING routine, 2-42  
 DISPLAY WIDTH MM routine, 2-44  
 DISPLAY WIDTH routine, 2-43  
 DOES BACKING STORE routine, 2-44  
 DOES SAVE UNDERS routine, 2-45  
 Drawable  
   copying rectangular area from, 2-23  
   obtaining geometry, 2-75  
 DRAW ARC routine, 2-45  
 DRAW ARCS routine, 2-46  
 DRAW IMAGE STRING 16 routine, 2-47  
 DRAW IMAGE STRING routine, 2-46  
 Drawing  
   arcs, 2-46  
   image text string (16-bit), 2-47  
   image text string (8-bit), 2-46  
   points, 2-49  
   rectangles, 2-50  
   text, 2-52  
   text string (16-bit), 2-52  
   text string (8-bit), 2-51  
 Drawing arc, 2-45  
 Drawing rectangle  
   multiple, 2-50  
 Drawing segment, 2-51  
 DRAW LINE routine, 2-47  
 DRAW LINES routine, 2-48  
 DRAW POINT routine, 2-49  
 DRAW POINTS routine, 2-49  
 DRAW RECTANGLE routine, 2-50  
 DRAW RECTANGLES routine, 2-50  
 DRAW SEGMENTS routine, 2-51  
 DRAW STRING 16 routine, 2-52  
 DRAW STRING routine, 2-51  
 DRAW TEXT 16 routine, 2-53  
 DRAW TEXT routine, 2-52  
 DXmActivateWidget routine, 8-1



DXmChangeWindowGeometry routine, 8-1  
 DXmChildren routine, 8-2  
 DXmColorMixCallbackStruct data structure, 9-1  
 DXmColorMixGetNewColor routine, 8-2  
 DXmColorMixSetNewColor routine, 8-3  
 DXmCreateColorMixDialog routine, 8-4  
 DXmCreateColorMix routine, 8-3  
 DXmCreateCSText routine, 8-4  
 DXmCreateCursor routine, 8-4  
 DXmCreateHelpDialog routine, 8-5  
 DXmCreateHelp routine, 8-5  
 DXmCreatePrintBox routine, 8-6  
 DXmCreatePrintDialog routine, 8-6  
 DXmCreateScrolledCSText routine, 8-6  
 DXmCreateSvn routine, 8-7  
 DXmCSContainsStringCharSet routine, 8-7  
 DXmCSTextCallbackStruct data structure, 9-1  
 DXmCSTextClearSelection routine, 8-8  
 DXmCSTextCopy routine, 8-8  
 DXmCSTextCut routine, 8-8  
 DXmCSTextDisableRedisplay routine, 8-9  
 DXmCSTextEnableRedisplay routine, 8-9  
 DXmCSTextGetEditable routine, 8-9  
 DXmCSTextGetInsertionPosition routine, 8-10, 8-15  
 DXmCSTextGetLastPosition routine, 8-10  
 DXmCSTextGetMaxLength routine, 8-11  
 DXmCSTextGetSelectionInfo routine, 8-11  
 DXmCSTextGetSelection routine, 8-11  
 DXmCSTextGetString routine, 8-12  
 DXmCSTextGetStringWrapped routine, 8-12  
 DXmCSTextGetTopPath routine, 8-12  
 DXmCSTextGetTopPosition routine, 8-13  
 DXmCSTextHasSelection routine, 8-13  
 DXmCSTextHorizontalScroll routine, 8-14  
 DXmCSTextInsert routine, 8-14  
 DXmCSTextInvalidate routine, 8-15  
 DXmCSTextMarkRedraw routine, 8-14  
 DXmCSTextNumLines routine, 8-15  
 DXmCSTextPaste routine, 8-16  
 DXmCSTextPosToLine routine, 8-16  
 DXmCSTextPosToXY routine, 8-17  
 DXmCSTextRead routine, 8-17  
 DXmCSTextRemove routine, 8-17  
 DXmCSTextReplace routine, 8-18  
 DXmCSTextSetAddMode routine, 8-18  
 DXmCSTextSetEditable routine, 8-19  
 DXmCSTextSetHighlight routine, 8-19  
 DXmCSTextSetInsertionPosition routine, 8-19  
 DXmCSTextSetMaxLength routine, 8-20  
 DXmCSTextSetSelection routine, 8-20  
 DXmCSTextSetString routine, 8-20  
 DXmCSTextSetTopPosition routine, 8-21  
 DXmCSTextShowPosition routine, 8-21  
 DXmCSTextVerifyCallbackStruct data structure, 9-2  
 DXmCSTextVerticalScroll routine, 8-22  
 DXmCSTextXYToPos routine, 8-22  
 DXmCvtCStoDDIF routine, 8-22  
 DXmCvtCStoFC routine, 8-23  
 DXmCvtCStoOS routine, 8-23  
 DXmCvtDDIFtoCS routine, 8-24  
 DXmCvtFCtoCS routine, 8-24  
 DXmCvtOStoCS routine, 8-24  
 DXmDisplayCSMessage routine, 8-25  
 DXmDisplayVmsMessage, 8-25  
 DXmFindFontFallback routine, 8-26  
 DXmFontListCreateDefault routine, 8-26  
 DXmFormSpaceButtonsEqually routine, 8-27  
 DXmGetLocaleCharset routine, 8-27  
 DXmGetLocaleCharsets routine, 8-28  
 DXmGetLocaleMnemonic routine, 8-28  
 DXmGetLocaleString routine, 8-28  
 DXmHelpOnContext routine, 8-29  
 DXmHelpSystemClose routine, 8-29  
 DXmHelpSystemDisplay routine, 8-29  
 DXmHelpSystemOpen routine, 8-30  
 DXmInitialize routine, 8-30  
 DXmLoadQueryFont routine, 8-30  
 DXmMakeGeometryRequest routine, 8-2  
 DXmNumChildren routine, 8-31  
 DXmPositionWidget routine, 8-31  
 DXmPrintFormatStruct data structure, 9-2  
 DXmPrintOptionsMenuStruct data structure, 9-2  
 DXmPrintWgtAugmentList routine, 8-32  
 DXmPrintWgtPrintJob routine, 8-32  
 DXmStringCheck routine, 8-32  
 DXmSvnAddEntries routine, 8-33  
 DXmSvnAutoScrollCheck routine, 8-33  
 DXmSvnAutoScrollDisplay routine, 8-34  
 DXmSvnCallbackStruct data structure, 9-3  
 DXmSvnClearHighlighting routine, 8-34  
 DXmSvnClearHighlight routine, 8-34  
 DXmSvnClearSelection routine, 8-35  
 DXmSvnClearSelections routine, 8-35  
 DXmSvnDeleteEntries routine, 8-35  
 DXmSvnDisableDisplay routine, 8-36  
 DXmSvnEnableDisplay routine, 8-36  
 DXmSvnFlushEntry routine, 8-36  
 DXmSvnGetComponentNumber routine, 8-37  
 DXmSvnGetComponentTag routine, 8-37  
 DXmSvnGetComponentText routine, 8-38  
 DXmSvnGetComponentWidth routine, 8-37  
 DXmSvnGetDisplayed routine, 8-38  
 DXmSvnGetEntryLevel routine, 8-39  
 DXmSvnGetEntryNumber routine, 8-39  
 DXmSvnGetEntryPosition routine, 8-39  
 DXmSvnGetEntrySensitivity routine, 8-40  
 DXmSvnGetEntryTag routine, 8-40  
 DXmSvnGetHighlighted routine, 8-41  
 DXmSvnGetNumDisplayed routine, 8-41  
 DXmSvnGetNumHighlighted routine, 8-41

DXmSvnGetNumSelections routine, 8-42  
 DXmSvnGetPrimaryWorkWidget routine, 8-42  
 DXmSvnGetSecondaryWorkWidget routine, 8-43  
 DXmSvnGetSelections routine, 8-43  
 DXmSvnGetTreePosition routine, 8-43  
 DXmSvnHideHighlighting routine, 8-44  
 DXmSvnHideSelections routine, 8-44  
 DXmSvnHighlightAll routine, 8-44  
 DXmSvnHighlightEntry routine, 8-45  
 DXmSvnInsertComponent routine, 8-45  
 DXmSvnInvalidateEntry routine, 8-46  
 DXmSvnMapPosition routine, 8-46  
 DXmSvnPositionDisplay routine, 8-46  
 DXmSvnRemoveComponent routine, 8-47  
 DXmSvnSelectAll routine, 8-47  
 DXmSvnSelectComponent routine, 8-48  
 DXmSvnSelectEntry routine, 8-48  
 DXmSvnSetApplDragging routine, 8-48  
 DXmSvnSetComponentHidden routine, 8-49  
 DXmSvnSetComponentPixmap routine, 8-49  
 DXmSvnSetComponentTag routine, 8-50  
 DXmSvnSetComponentText routine, 8-50  
 DXmSvnSetComponentWidget routine, 8-51  
 DXmSvnSetComponentWidth routine, 8-51  
 DXmSvnSetEntryIndexWindow routine, 8-52  
 DXmSvnSetEntryNumComponents routine, 8-52  
 DXmSvnSetEntryPosition routine, 8-53  
 DXmSvnSetEntry routine, 8-51  
 DXmSvnSetEntrySensitivity routine, 8-53  
 DXmSvnSetEntryTag routine, 8-54  
 DXmSvnSetTreePosition routine, 8-54  
 DXmSvnShowHighlighting routine, 8-54  
 DXmSvnShowSelections routine, 8-55  
 DXmSvnValidateAll routine, 8-55  
 DXmSvnWidget routine, 8-55

## E

EMPTY REGION routine, 2-54  
 ENABLE ACCESS CONTROL routine, 2-54  
 Endpoint patterns  
   changing, 2-186  
 Entry in color map  
   determining color, 2-142  
 EQUAL REGION routine, 2-54  
 Error codes  
   list of, 3-68  
 Error event data structure, 3-17, 3-18  
 Error handler  
   description, 3-67  
   nonfatal, 2-180  
   specifying nonfatal handler, 2-180  
 Error handling, 3-67  
 Event  
   allowing, 2-7  
   checking number queued, 2-55  
   checking type of, 2-17  
   enabling synchronization, 2-209

Event (cont'd)  
   flushing, 2-62  
   getting next, 2-129  
   matching window and type, 2-17  
   peeking at, 2-133  
   putting back on queue, 2-137  
   removing next matching, 2-125  
   returning pending, 2-134  
   selecting asynchronous input types, 2-172  
   selecting asynchronous types, 2-172  
   selecting input types, 2-173  
   sending to client, 2-173  
   specifying synchronization handler, 2-175  
 Event data structure, 3-18  
 Event mask, 3-19  
   in CHANGE ACTIVE POINTER GRAB routine,  
     2-12  
   initial root, 2-55  
   obtaining, 2-55  
   table, 3-19  
 EVENT MASK OF SCREEN routine, 2-55  
 Event routines  
   CHECK IF EVENT, 2-16  
   CHECK MASK EVENT, 2-16  
   CHECK TYPED EVENT, 2-17  
   CHECK WINDOW EVENT, 2-18  
   EVENTS QUEUED, 2-55  
   FLUSH, 2-62  
   GET ERROR DATABASE TEXT, 2-72  
   GET ERROR TEXT, 2-73  
   GET MOTION EVENTS, 2-82  
   IF EVENT, 2-107  
   MASK EVENT, 2-125  
   NEXT EVENT, 2-129  
   PEEK EVENT, 2-133  
   PEEK IF EVENT, 2-133  
   PENDING, 2-134  
   PUT BACK EVENT, 2-137  
   SELECT ASYNC EVENT, 2-172  
   SELECT ASYNC INPUT, 2-172  
   SELECT INPUT, 2-173  
   SEND EVENT, 2-173  
   SET AFTER FUNCTION, 2-175  
   SET ERROR HANDLER, 2-180  
   SET IO ERROR HANDLER, 2-186  
   SYNC, 2-208  
   SYNCHRONIZE, 2-209  
   WINDOW EVENT, 2-222  
 Event Routines  
   FILTER EVENT, 2-61  
 EVENTS QUEUED routine, 2-55  
 Expose event data structure, 3-19  
 Exposures  
   allowing, 2-190  
   obtaining values, 2-87  
 Ext codes data structure, 3-21

Ext data data structure, 3-21

## F

---

Fatal error

handling, 2-186

FETCH BUFFER routine, 2-56

FETCH BYTES routine, 2-56

FETCH NAME routine, 2-57

FILL ARC routine, 2-58

FILL ARCS routine, 2-58

Filling multiple rectangles, 2-60

Filling rectangle, 2-60

FILL POLYGON routine, 2-59

FILL RECTANGLE routine, 2-60

FILL RECTANGLES routine, 2-60

Fill style

changing, 2-180, 2-181

FILTER EVENT routine, 2-61

FIND CONTEXT routine, 2-61

FLUSH GC routine, 2-62

FLUSH routine, 2-62

Focus change event data structure, 3-22

Focus window

obtaining identifier, 2-80

Font

changing value in graphics context, 2-182

freeing storage of, 2-65

getting directory path, 2-73

listing, 2-113, 2-114

loading, 2-121

properties, 2-74

returning information about, 2-121, 2-142

setting the directory path, 2-182

unloading, 2-218

Font characters, 2-69

Font data structure, 3-24

Font property data structure, 3-23

Font Routines

FREE FONT, 2-65

GET CHAR STRUCT, 2-69

GET FONT PATH, 2-73

GET FONT PROPERTY, 2-74

GET ICON NAME, 2-77

LIST FONT, 2-113

LIST FONTS, 2-114

LIST FONTS WITH INFO, 2-116

LIST FONT WITH INFO, 2-114

LOAD FONT, 2-121

LOAD QUERY FONT, 2-121

QUERY FONT, 2-142

SET FONT PATH, 2-182

UNLOAD FONT, 2-218

FORCE SCREEN SAVER routine, 2-62

Foreground color

changing in graphics context, 2-182, 2-193

Format of Xlib documentation, 1-1 to 1-5

Format section, 1-1, 1-5

FREE COLORMAP routine, 2-63

FREE COLORS routine, 2-64

FREE CURSOR routine, 2-64

FREE FONT routine, 2-65

FREE GC routine, 2-65

Freeing color map, 2-23

FREE MODIFIER MAP routine, 2-65

FREE PIXMAP routine, 2-66

FREE routine, 2-63

FREE STRING LIST routine, 2-66

Function

changing in graphics context, 2-193

changing value, 2-183

## G

---

GC

See Graphics context

GC mask, 3-28

GCONTEXT FROM GC routine, 2-66

Geometry

obtaining for specified drawable, 2-75

parsing, 2-132

parsing window, 2-67

GEOMETRY routine, 2-67

GET ATOM NAME routine, 2-68

GET CHAR STRUCT routine, 2-69

GET CLASS HINT routine, 2-70

GET COMMAND routine, 2-71

GET DEFAULT routine, 2-71

GET ERROR DATABASE TEXT routine, 2-72

GET ERROR TEXT routine, 2-73

GET FONT PATH routine, 2-73

GET FONT PROPERTY routine, 2-74

GET GC VALUES routine, 2-75

GET GEOMETRY routine, 2-75

GET ICON NAME routine, 2-77

GET ICON SIZES routine, 2-78

GET IMAGE routine, 2-79

GET INPUT FOCUS routine, 2-80

GET KEYBOARD CONTROL routine, 2-80

GET KEYBOARD MAPPING routine, 2-81

GET MODIFIER MAPPING routine, 2-82

GET MOTION EVENTS routine, 2-82

GET NORMAL HINTS routine, 2-83

GET PIXEL routine, 2-84

GET POINTER CONTROL routine, 2-84

GET POINTER MAPPING routine, 2-85

GET RGB COLORMAPS routine, 2-86

GET SCREEN SAVER routine, 2-87

GET SELECTION OWNER routine, 2-88

GET SIZE HINTS routine, 2-88

GET STANDARD COLORMAP routine, 2-88

GET SUB IMAGE routine, 2-89

GET TEXT PROPERTY routine, 2-90  
 GET TRANSIENT FOR HINT routine, 2-91  
 GET VISUAL INFO routine, 2-91  
 GET WINDOW ATTRIBUTES routine, 2-92  
 GET WINDOW PROPERTY routine, 2-93  
 GET WM CLIENT MACHINE routine, 2-94  
 GET WM COLORMAP WINDOWS routine, 2-95  
 GET WM HINTS routine, 2-96  
 GET WM ICON NAME routine, 2-96  
 GET WM NAME routine, 2-97  
 GET WM NORMAL HINTS routine, 2-97  
 GET WM PROTOCOLS routine, 2-98  
 GET ZOOM HINTS routine, 2-99  
 Grabbing  
   button, 2-100  
   changing active pointer, 2-11  
   key, 2-102  
   keyboard, 2-103  
   pointer, 2-104  
 Grabbing server, 2-106  
 GRAB BUTTON routine, 2-100  
   See also UNGRAB BUTTON routine  
 GRAB KEYBOARD routine, 2-103  
   See also UNGRAB KEYBOARD routine  
 GRAB KEY routine, 2-102  
   See also UNGRAB KEY routine  
 GRAB POINTER routine, 2-104  
   See also UNGRAB POINTER routine  
 GRAB SERVER routine, 2-106  
   See also UNGRAB SERVER routine  
 Graphics context data structure, 3-25  
 Graphics Context Routines  
   CHANGE GC, 2-12  
   COPY GC, 2-24  
   CREATE GC, 2-26  
   DEFAULT GC, 2-32  
   FLUSH GC, 2-62  
   GCONTEXT FROM GC, 2-66  
   GET GC VALUES, 2-75  
   QUERY BEST STIPPLE, 2-140  
   QUERY BEST TILE, 2-141  
   SET ARC MODE, 2-175  
   SET BACKGROUND, 2-176  
   SET CLIP MASK, 2-177  
   SET CLIP ORIGIN, 2-177  
   SET CLIP RECTANGLES, 2-178  
   SET DASHES, 2-179  
   SET FILL RULE, 2-180  
   SET FILL STYLE, 2-181  
   SET FOREGROUND, 2-182  
   SET FUNCTION, 2-183  
   SET GRAPHICS EXPOSURES, 2-184  
   SET LINE ATTRIBUTES, 2-186  
   SET PLANE MASK, 2-188  
   SET STATE, 2-193  
   SET STIPPLE, 2-194  
   SET SUBWINDOW MODE, 2-194

#### Graphics Context Routines (cont'd)

  SET TILE, 2-195  
   SET TS ORIGIN, 2-196  
 Graphics expose event data structure, 3-29  
 Graphics Routines  
   ADD PIXEL, 2-2  
   CLEAR AREA, 2-19  
   CLEAR WINDOW, 2-20  
   COPY AREA, 2-23  
   COPY PLANE, 2-24  
   CREATE IMAGE, 2-27  
   DESTROY IMAGE, 2-37  
   DRAW ARC, 2-45  
   DRAW ARCS, 2-46  
   DRAW LINE, 2-47  
   DRAW LINES, 2-48  
   DRAW POINT, 2-49  
   DRAW POINTS, 2-49  
   DRAW RECTANGLE, 2-50  
   DRAW RECTANGLES, 2-50  
   DRAW SEGMENTS, 2-51  
   FILL ARC, 2-58  
   FILL ARCS, 2-58  
   FILL POLYGON, 2-59  
   FILL RECTANGLE, 2-60  
   FILL RECTANGLES, 2-60  
   FREE GC, 2-65  
   GET IMAGE, 2-79  
   GET PIXEL, 2-84  
   GET SUB IMAGE, 2-89  
   PUT IMAGE, 2-137  
   PUT PIXEL, 2-138  
   QUERY BEST SIZE, 2-139  
   SET FONT, 2-182  
   SUB IMAGE, 2-208  
 Gravity event data structure, 3-30

## H

---

HEIGHT MM OF SCREEN routine, 2-106  
 Height of screen, 2-39, 2-40, 2-106  
 Host  
   adding multiple to connect list, 2-2  
   adding single to connect list, 2-1  
   listing accessible, 2-117  
   removing from access list, 2-152  
 Host address data structure, 3-31

## I

---

I/O error handler, 2-186  
 Icon  
   getting name, 2-77  
   getting recommended sizes, 2-78  
   setting recommended sizes, 2-185  
   specifying name, 2-184

ICONIFY WINDOW routine, 2-107  
 Icon size data structure, 3-32  
 IF EVENT routine, 2-107  
     See also CHECK IF EVENT routine  
 Image  
     allocating memory for image, 2-27  
     copying a portion of existing image, 2-208  
     deallocating memory for image, 2-37  
     obtaining, 2-79, 2-137  
     obtaining pixel value, 2-84  
     setting pixel value, 2-138  
 IMAGE BYTE ORDER routine, 2-108  
 Image data structure, 3-33  
 Image text string (16-bit)  
     drawing, 2-47  
 Image text string (8-bit)  
     drawing, 2-46  
 Information routines  
     GET CHAR STRUCT, 2-69  
     GET FONT PROPERTY, 2-74  
     LIST FONT, 2-113  
     LIST FONTS, 2-114  
     LIST FONT WITH INFO, 2-114  
 Initializing  
     resource manager, 2-159  
 Input event queue  
     obtaining length, 2-138  
 Input focus  
     changing, 2-185  
     obtaining, 2-80  
 INSERT MODIFIERMAP ENTRY routine, 2-108  
 INSTALL COLORMAP routine, 2-109  
     See also UNINSTALL COLORMAP routine  
 INTERN ATOM routine, 2-110  
 Intersection  
     computing for region, 2-110  
 INTERSECT REGION routine, 2-110  
 Intrinsic Data Structures  
     Arg, 5-1  
     XtActionsRec, 5-1  
     XtCallbackRec, 5-1  
     XtConvertArgRec, 5-2  
     XtI18nContextRec, 5-2  
     XtPopdownIdRec, 5-2  
     XtResource, 5-2  
     XtSubstitutionRec, 5-3  
     XtWidgetGeometry, 5-3

## J

---

Join style  
     changing, 2-186

## K

---

Key  
     passively grabbing, 2-102  
     releasing from grab, 2-215  
 Keyboard  
     changing mapping, 2-13  
     changing settings, 2-13  
     grabbing control, 2-103  
     obtaining logical state, 2-143  
     obtaining mapping, 2-81  
     obtaining modifier keys, 2-82  
     obtaining settings, 2-80  
     releasing active grab, 2-215  
     setting modifier keys, 2-187  
 Keyboard bell  
     ringing, 2-9  
 Keyboard control data structure, 3-36  
 Keyboard keys  
     turning off auto-repeat, 2-8  
     turning on auto-repeat, 2-8  
 Keyboard mapping  
     refreshing, 2-151  
 Keyboard state data structure, 3-37  
 Key click, 2-13  
     obtaining value, 2-80  
 Key code, 2-81, 2-82  
     converting from key symbol, 2-111, 2-207  
     converting to key symbol, 2-110  
     returning corresponding key symbol, 2-123  
     setting key symbols, 2-13  
 Key codes  
     setting modifier keys, 2-187  
 KEYCODE TO KEYSYM routine, 2-110  
     See also KEYSYM TO KEYCODE routine  
 Key event  
     mapping to string, 2-123  
 Key event data structure, 3-35  
 Key map  
     querying, 2-143  
 Keymap event data structure, 3-38  
 Key symbol  
     converting from key code, 2-110  
     converting to key code, 2-111, 2-207  
     converting to string, 2-111  
     looking up, 2-123  
     obtaining mapping, 2-81  
     rebinding to string, 2-149  
     specifying, 2-13  
 KEYSYM TO KEYCODE routine, 2-111  
     See also KEYCODE TO KEYSYM routine  
 KEYSYM TO STRING routine, 2-111  
     See also STRING TO KEYSYM routine  
 KILL CLIENT routine, 2-112



## L

---

LAST KNOWN REQUEST PROCESSED routine, 2-112

LED, 2-13

- obtaining value, 2-80

Line

- drawing, 2-47
- drawing connected lines, 2-48
- drawing unconnected lines, 2-51

Line style

- changing values, 2-186

Line width

- changing, 2-186

LIST DEPTHS routine, 2-113

LIST FONT routine, 2-113

LIST FONTS routine, 2-114

LIST FONTS WITH INFO routine, 2-116

LIST FONT WITH INFO routine, 2-114

LIST HOSTS routine, 2-117

LIST INSTALLED COLORMAPS routine, **2-118**

LIST PIXMAP FORMATS routine, 2-119

LIST PROPERTIES routine, 2-120

LOAD FONT routine, 2-121

Loading

- database, 2-160

LOAD QUERY FONT routine, 2-121

LOOKUP COLOR routine, 2-122

LOOKUP KEYSYM routine, 2-123

LOOKUP STRING routine, 2-123

- See also REBIND KEYSYM routine

Lowering a window, 2-124

LOWER WINDOW routine, 2-124

## M

---

Map event data structure, 3-39

Mapping all subwindows, 2-124

Mapping a window, 2-124

Mapping event data structure, 3-40

Mapping windows and all subwindows, 2-125

MAP RAISED routine, 2-124

Map request event data structure, 3-39

MAP SUBWINDOWS routine, 2-124

MAP WINDOW routine, 2-125

Mask event, 3-19

MASK EVENT routine, 2-125

- See also CHECK MASK EVENT routine

MATCH VISUAL INFO routine, 2-125

MAX CMAPS OF SCREEN routine, 2-126

MAX REQUEST SIZE routine, 2-126

Mechanism, 1-5

- passing, 1-11

Merging

- databases, 2-159

MIN CMAPS OF SCREEN routine, 2-127

Modifier keymap data structure, 3-41

Modifier keys

- obtaining key code, 2-82
- setting, 2-187

Modifier map

- freeing storage, 2-65
- inserting entry, 2-108
- setting modifier keys, 2-187

MotifWmHints data structure, 7-1

MotifWmInfo data structure, 7-1

Motion event data structure, 3-42

Mouse

- getting motion event, 2-82

MOVE RESIZE WINDOW routine, 2-127

MOVE WINDOW routine, 2-127

Moving a window, 2-127

MrmCloseHierarchy routine, 6-1

MrmFetchBitmapLiteral routine, 6-1

MrmFetchColorLiteral routine, 6-2

MrmFetchIconLiteral routine, 6-2

MrmFetchInterfaceModule routine, 6-3

MrmFetchLiteral routine, 6-3

MrmFetchSetValues routine, 6-4

MrmFetchWidgetOverride routine, 6-5

MrmFetchWidget routine, 6-4

MrmInitialize routine, 6-5

MrmOpenHierarchyPerDisplay routine, 6-6

MrmOpenHierarchy routine, 6-5

MrmOsOpenParam data structure, 7-1

MrmRegisterArg data structure, 7-2

MrmRegisterClass routine, 6-6

MrmRegisterNamesInHierarchy routine, 6-7

MrmRegisterNames routine, 6-7

## N

---

NEW MODIFIERMAP routine, 2-128

NEXT EVENT routine, 2-129

NEXT REQUEST routine, 2-129

No expose event data structure, 3-43

Nonfatal error handler, 2-180

NO OP routine, 2-129

## O

---

OFFSET REGION routine, 2-130

OPEN DISPLAY routine, 2-130

Origin

- tile or stipple, 2-196

## P

---

Parent window

- changing, 2-152

PARSE COLOR routine, 2-131

PARSE GEOMETRY routine, 2-132  
 Passing mechanism  
   by descriptor, 1-5  
   by pointer, 1-5  
   by reference, 1-11  
   by value, 1-5, 1-11  
 Passive grab  
   deactivating for pointer button press, 2-214  
 PEEK EVENT routine, 2-133  
 PEEK IF EVENT routine, 2-133  
 PENDING routine, 2-134  
 PERM ALLOC routine, 2-134  
 Pixel  
   black, 2-11  
   obtaining value from image, 2-84  
   setting value for image, 2-138  
   white, 2-221  
 Pixel value, 2-10  
   white, 2-220  
 Pixmap  
   creating, 2-28  
   creating from bitmap data, 2-29  
   freeing storage, 2-66  
 Pixmap and Bitmap Routines  
   CREATE BITMAP FROM DATA, 2-25  
   CREATE PIXMAP, 2-28  
   CREATE PIXMAP FROM BITMAP DATA,  
     2-29  
   FREE PIXMAP, 2-66  
   LIST PIXMAP FORMATS, 2-119  
   READ BITMAP FILE, 2-147  
   WRITE BITMAP FILE, 2-223  
 Pixmap format values data structure, 3-44  
 Plane  
   allocating, 2-5  
   changing plane mask, 2-188  
   obtaining number in screen, 2-134  
   screen, 2-42  
 Plane mask  
   changing in graphics context, 2-193  
 PLANES OF SCREEN routine, 2-134  
 Point  
   drawing, 2-49  
   drawing multiple, 2-49  
 Point data structure, 3-45  
 Pointer  
   acceleration of, 2-14, 2-84  
   actively grabbing, 2-104  
   changing active pointer grab, 2-11  
   deactivating passive grab, 2-214  
   defining movement, 2-14  
   moving, 2-220  
   obtaining coordinates, 2-143  
   obtaining mapping, 2-85  
   obtaining movement values, 2-84  
   obtaining root window, 2-143  
   passive grabbing, 2-100  
   releasing active grab, 2-216  
   Pointer (cont'd)  
     setting mapping, 2-189  
     threshold speed of, 2-14  
   Pointer mapping  
     obtaining list, 2-85  
   POINT IN REGION routine, 2-135  
 Polygon  
   filling, 2-59  
 POLYGON REGION routine, 2-135  
 Property  
   changing, 2-14  
   deleting, 2-37  
   getting window manager hints, 2-96  
   icon sizes, 2-78, 2-185  
   obtaining format, 2-93  
   obtaining list, 2-120  
   obtaining type, 2-93  
   regular window sizes, 2-83, 2-188  
   rotating, 2-169  
   setting command, 2-179  
   setting standard, 2-192  
   setting window manager hints, 2-200  
   zoom window sizes, 2-99  
 Property event data structure, 3-45  
 Property Routines  
   ALLOC CLASS HINT, 2-3  
   ALLOC ICON SIZE, 2-5  
   ALLOC STANDARD COLORMAP, 2-6  
   CHANGE PROPERTY, 2-14  
   CONVERT SELECTION, 2-22  
   DELETE CONTEXT, 2-35  
   DELETE PROPERTY, 2-37  
   FETCH BUFFER, 2-56  
   FETCH BYTES, 2-56  
   FETCH NAME, 2-57  
   FIND CONTEXT, 2-61  
   FREE STRING LIST, 2-66  
   GET ATOM NAME, 2-68  
   GET CLASS HINT, 2-70  
   GET COMMAND, 2-70  
   GET ICON SIZES, 2-78  
   GET NORMAL HINTS, 2-83  
   GET SELECTION OWNER, 2-88  
   GET SIZE HINTS, 2-88  
   GET TEXT PROPERTY, 2-90  
   GET TRANSIENT FOR HINT, 2-91  
   GET WINDOW PROPERTY, 2-93  
   GET WM CLIENT MACHINE, 2-94  
   GET WM HINTS, 2-96  
   GET WM ICON NAME, 2-96  
   GET WM NAME, 2-97  
   GET WM NORMAL HINTS, 2-97  
   GET WM PROTOCOLS, 2-98  
   GET ZOOM HINTS, 2-99  
   INTERN ATOM, 2-110  
   LIST PROPERTIES, 2-120  
   ROTATE WINDOW PROPERTIES, 2-169  
   SAVE CONTEXT, 2-170

## Property Routines (cont'd)

- SET CLASS HINT, 2-176
- SET COMMAND, 2-179
- SET ICON NAME, 2-184
- SET ICON SIZES, 2-185
- SET NORMAL HINTS, 2-188
- SET SELECTION OWNER, 2-191
- SET SIZE HINTS, 2-191
- SET STANDARD PROPERTIES, 2-192
- SET TEXT PROPERTY, 2-195
- SET TRANSIENT FOR HINT, 2-195
- SET WM CLIENT MACHINE, 2-199
- SET WM HINTS, 2-200
- SET WM ICON NAME, 2-200
- SET WM NAME, 2-200
- SET WM NORMAL HINTS, 2-201
- SET WM PROPERTIES, 2-201
- SET WM PROTOCOLS, 2-202
- SET WM SIZE HINTS, 2-202
- SET ZOOM HINTS, 2-203
- STORE NAME, 2-206
- STRING LIST TO TEXT PROPERTY, 2-207
- TEXT PROPERTY TO STRING LIST, 2-211
- UNIQUE CONTEXT, 2-218

PropMotifWmHints data structure, 7-2

PropMotifWmInfo data structure, 7-2

Protocol revision number

- obtaining, 2-136

PROTOCOL REVISION routine, 2-136

Protocol version number

- obtaining, 2-136

PROTOCOL VERSION routine, 2-136

Pseudocolor

- allocating entries and planes, 2-4

PUT BACK EVENT routine, 2-137

PUT IMAGE routine, 2-29, 2-137

PUT PIXEL routine, 2-138

## Q

---

Q LENGTH routine, 2-138

Quark

- allocating, 2-168
- converting to string, 2-166

QUERY BEST CURSOR routine, 2-138

QUERY BEST SIZE routine, 2-139

QUERY BEST STIPPLE routine, 2-140

QUERY BEST TILE routine, 2-141

QUERY COLOR routine, 2-141

QUERY COLORS routine, 2-142

QUERY FONT routine, 2-142

QUERY KEYMAP routine, 2-143

QUERY POINTER routine, 2-143

QUERY TEXT EXTENTS 16 routine, 2-145

QUERY TEXT EXTENTS routine, 2-144

QUERY TREE routine, 2-146

## R

---

RAISE WINDOW routine, 2-147

Raising a window, 2-124, 2-147

READ BITMAP FILE routine, 2-147

REBIND KEYSYM routine, 2-149

- See also LOOKUP STRING routine

RECOLOR CURSOR routine, 2-149

RECONFIGURE WM WINDOW routine, 2-150

Rectangle

- drawing, 2-50
- drawing multiple, 2-50
- filling, 2-60
- filling multiple, 2-60
- generating smallest enclosing, 2-20

Rectangle data structure, 3-46

Rectangle location within region, 2-150

RECT IN REGION routine, 2-150

Reference passing mechanism, 1-5, 1-11

REFRESH KEYBOARD MAPPING routine, 2-151

Region

- calculating the union, 2-217
- calculating the union rectangle, 2-217
- computing the intersection, 2-110
- creating, 2-135
- creating empty region, 2-29
- deallocating storage, 2-37
- destroying region, 2-37
- determining if empty, 2-54
- determining if equal, 2-54
- determining location of rectangle, 2-150
- determining point location, 2-135
- expanding size of, 2-203
- moving by offset, 2-130
- reducing size of, 2-203
- setting graphics context for, 2-189
- subtracting regions, 2-208

Region Routines

- CLIB BOX, 2-20
- CREATE REGION, 2-29
- DESTROY REGION, 2-37
- EMPTY REGION, 2-54
- EQUAL REGION, 2-54
- INTERSECT REGION, 2-110
- OFFSET REGION, 2-130
- POINT IN REGION, 2-135
- POLYGON REGION, 2-135
- RECOLOR CURSOR, 2-150
- SET REGION, 2-189
- SHRINK REGION, 2-203
- SUBTRACT REGION, 2-208
- UNION RECT WITH REGION, 2-217
- UNION REGION, 2-217
- XOR REGION, 2-224



REMOVE FROM SAVE SET routine, 2-151  
     See also ADD TO SAVE SET routine  
 REMOVE HOST routine, 2-152  
     See also REMOVE HOSTS routine  
 REMOVE HOSTS routine, 2-152  
     See also REMOVE HOST routine  
 Reparent event data structure, 3-47  
 REPARENT WINDOW routine, 2-152  
 Replacing color map, 2-198  
 RESET SCREEN SAVER routine, 2-153  
 Resize request event data structure, 3-48  
 RESIZE WINDOW routine, 2-153  
 Resizing a window, 2-127, 2-153  
 Resource  
     adding, 2-162, 2-165  
     retrieving, 2-157, 2-162  
     searching, 2-164  
     storing, 2-161, 2-165  
 Resource entry  
     adding, 2-161  
 Resource manager  
     initializing, 2-159  
 Resource manager option data structure, 3-49  
 Resource Manager Routines  
     PERM ALLOC, 2-134  
     RM COMBINE DATABASE, 2-155  
     RM COMBINE FILE DATABASE, 2-155  
     RM DESTROY DATABASE, 2-155  
     RM ENUMERATE DATABASE, 2-156  
     RM GET DATABASE, 2-156  
     RM GET FILE DATABASE, 2-157  
     RM GET RESOURCE, 2-157  
     RM GET STRING DATABASE, 2-158  
     RM INITIALIZE, 2-159  
     RM LOCALE OF DATABASE, 2-159  
     RM MERGE DATABASES, 2-159  
     RM PARSE COMMAND, 2-160  
     RM PERM STRING TO QUARK, 2-160  
     RM PUT FILE DATABASE, 2-161  
     RM PUT LINE RESOURCE, 2-161  
     RM PUT RESOURCE, 2-161  
     RM PUT STRING RESOURCE, 2-162  
     RM Q GET RESOURCE, 2-162  
     RM Q GET SEARCH LIST, 2-164  
     RM Q GET SEARCH RESOURCE, 2-164  
     RM Q PUT RESOURCE, 2-165  
     RM Q PUT STRING RESOURCE, 2-165  
     RM QUARK TO STRING, 2-166  
     RM SET DATABASE, 2-166  
     RM STRING TO BIND QUARK LIST, 2-167  
     RM STRING TO QUARK, 2-167  
     RM STRING TO QUARK LIST, 2-167  
     RM UNIQUE QUARK, 2-168  
 RESOURCE MANAGER STRING routine, 2-154  
 Resource manager value data structure, 3-49  
 Restacking array of windows, 2-154  
 RESTACK WINDOWS routine, 2-154  
 Retrieving  
     resource, 2-162  
 Retrieving database, 2-157  
 Retrieving resource, 2-157  
 RM COMBINE DATABASE routine, 2-155  
 RM COMBINE FILE DATABASE routine, 2-155  
 RM DESTROY DATABASE routine, 2-155  
 RM ENUMERATE DATABASE routine, 2-156  
 RM GET DATABASE routine, 2-156  
 RM GET FILE DATABASE routine, 2-157  
 RM GET RESOURCE routine, 2-157  
 RM GET STRING DATABASE routine, 2-158  
 RM INITIALIZE routine, 2-159  
 RM LOCALE OF DATABASE routine, 2-159  
 RM MERGE DATABASES routine, 2-159  
 RM PARSE COMMAND routine, 2-160  
 RM PERM STRING TO QUARK routine, 2-160  
 RM PUT FILE DATABASE routine, 2-161  
 RM PUT LINE RESOURCE routine, 2-161  
 RM PUT RESOURCE routine, 2-161  
 RM PUT STRING RESOURCE routine, 2-162  
 RM Q GET RESOURCE routine, 2-162  
 RM Q GET SEARCH LIST routine, 2-164  
 RM Q GET SEARCH RESOURCE routine, 2-164  
 RM Q PUT RESOURCE routine, 2-165  
 RM Q PUT STRING RESOURCE routine, 2-165  
 RM QUARK TO STRING routine, 2-166  
 RM SET DATABASE routine, 2-166  
 RM STRING TO BIND QUARK LIST routine, 2-167  
 RM STRING TO QUARK LIST routine, 2-167  
 RM STRING TO QUARK routine, 2-167  
 RM UNIQUE QUARK routine, 2-168  
 Root window  
     default, 2-33  
     obtaining default depth, 2-31  
     obtaining identifier, 2-168, 2-169  
 ROOT WINDOW OF SCREEN routine, 2-169  
 ROOT WINDOW routine, 2-168  
 ROTATE BUFFERS routine, 2-169  
 ROTATE WINDOW PROPERTIES routine, 2-169  
 Routine name section, 1-1, 1-5  
 Routine template, 1-5

## S

SAVE CONTEXT routine, 2-170  
 Save set  
     adding to, 2-2  
     removing windows from, 2-151  
 Saveset  
     changing, 2-15  
 Save unders, 2-45  
 Scanline, 2-10

- Screen
  - default, 2-33, 2-34
  - height, 2-39, 2-40, 2-106
  - obtaining default depth, 2-32
  - obtaining for specified display, 2-171
  - width, 2-43, 2-44, 2-221
- Screen blanking
  - obtaining values, 2-87
  - setting values for, 2-190
- SCREEN COUNT routine, 2-170
- SCREEN NUMBER OF SCREEN routine, 2-171
- SCREEN OF DISPLAY routine, 2-171
- SCREEN RESOURCE STRING routine, 2-171
- Screen saver
  - ACTIVATING, 2-1
  - forcing on, 2-62
  - invoking, 2-62
  - obtaining values, 2-87
  - resetting, 2-153
  - setting values for, 2-190
- Screen timeout period
  - obtaining values, 2-87
- Searching
  - resource, 2-164
- Segment data structure, 3-50
- SELECT ASYNC EVENT routine, 2-172
- SELECT ASYNC INPUT routine, 2-172
- SELECT INPUT routine, 2-173
- Selection clear event data structure, 3-50
- Selection event data structure, 3-51
- Selection request event data structure, 3-52
- SEND EVENT routine, 2-173
- Server
  - grabbing, 2-106
  - ungrabbing, 2-216
- Server release number, 2-219
- SERVER VENDOR routine, 2-174
- SET ACCESS CONTROL routine, 2-174
- SET AFTER FUNCTION routine, 2-175
- SET ARC MODE routine, 2-175
- SET BACKGROUND routine, 2-176
- SET CLASS HINT routine, 2-176
- SET CLIP MASK routine, 2-177
- SET CLIP ORIGIN routine, 2-177
- SET CLIP RECTANGLES routine, 2-178
- SET CLOSE DOWN MODE routine, 2-178
- SET COMMAND routine, 2-179
- SET DASHES routine, 2-179
- SET ERROR HANDLER routine, 2-180
- SET FILL RULE routine, 2-180
- SET FILL STYLE routine, 2-181
- SET FONT PATH routine, 2-182
- SET FONT routine, 2-182
- SET FOREGROUND routine, 2-182
- SET FUNCTION routine, 2-183
- SET GRAPHICS EXPOSURES routine, 2-184
- SET ICON NAME routine, 2-184
- SET ICON SIZES routine, 2-185
- SET INPUT FOCUS routine, 2-185
- SET IO ERROR HANDLER routine, 2-186
- SET LINE ATTRIBUTES routine, 2-186
- SET MODIFIER MAPPING routine, 2-187
- SET NORMAL HINTS routine, 2-188
- SET PLANE MASK routine, 2-188
- SET POINTER MAPPING routine, 2-189
- SET REGION routine, 2-189
- SET RGB COLORMAPS routine, 2-190
- SET SCREEN SAVER routine, 2-190
- SET SELECTION OWNER routine, 2-191
- SET SIZE HINTS routine, 2-191
- SET STANDARD COLORMAP routine, 2-192
- SET STANDARD PROPERTIES routine, 2-192
- SET STATE routine, 2-193
- SET STIPPLE routine, 2-194
- SET SUBWINDOW MODE routine, 2-194
- SET TEXT PROPERTY routine, 2-195
- SET TILE routine, 2-195
- SET TRANSIENT FOR HINT routine, 2-195
- SET TS ORIGIN routine, 2-196
- Set window attributes data structure, 3-53
- SET WINDOW BACKGROUND PIXMAP routine, 2-197
- SET WINDOW BACKGROUND routine, 2-196
- SET WINDOW BORDER PIXMAP routine, 2-197
- SET WINDOW BORDER routine, 2-197
- SET WINDOW BORDER WIDTH routine, 2-198
- SET WINDOW COLORMAP routine, 2-198
- SET WM CLIENT MACHINE routine, 2-199
- SET WM COLORMAP WINDOWS routine, 2-199
- SET WM HINTS routine, 2-200
- SET WM ICON NAME routine, 2-200
- SET WM NAME routine, 2-200
- SET WM NORMAL HINTS routine, 2-201
- SET WM PROPERTIES routine, 2-201
- SET WM PROTOCOLS routine, 2-202
- SET WM SIZE HINTS routine, 2-202
- SET ZOOM HINTS routine, 2-203
- Shift key
  - obtaining key code, 2-82
- SHRINK REGION routine, 2-203
- Shutdown
  - See Closing down
- Size hints data structure, 3-55
- Standard color map
  - obtaining, 2-88
  - setting, 2-192
- Standard color map data structure, 3-56
- Stipple
  - changing pixmap, 2-194
  - optimal size, 2-139, 2-140
- Stipple origin
  - changing in graphics context, 2-196

STORE BUFFER routine, 2-203  
 STORE BYTES routine, 2-204  
 STORE COLOR routine, 2-204  
 STORE COLORS routine, 2-205  
 STORE NAMED COLOR routine, 2-206  
 STORE NAME routine, 2-206  
 Storing  
     database, 2-161  
     resource, 2-161, 2-165  
 String  
     converting from key symbol, 2-111  
     converting to binding list, 2-167  
     converting to quark, 2-167  
     converting to quark list, 2-167  
     returning length, 2-212  
     returning logical extents, 2-209, 2-210  
 STRING LIST TO TEXT PROPERTY routine, 2-207  
 STRING TO KEYSYM routine, 2-207  
     See also KEYSYM TO STRING routine  
 Subimage  
     obtaining, 2-89  
 SUB IMAGE routine, 2-208  
 SUBTRACT REGION routine, 2-208  
 Subwindow  
     circulating down, 2-19  
     circulating in specified direction, 2-18  
     circulating up, 2-19  
     destroying, 2-38  
     mapping, 2-124  
     unmapping all, 2-218  
 Subwindow mode  
     changing, 2-194  
 Synchronization  
     disabling, 2-209  
     enabling, 2-209  
 Synchronization handler  
     specifying through SET AFTER FUNCTION routine, 2-175  
 SYNCHRONIZE routine, 2-209  
     See also SYNC routine  
 SYNC routine, 2-208  
     See also SYNCHRONIZE routine  
 Syntax, 1-1, 1-5

## T

---

Text  
     drawing, 2-52  
 TEXT EXTENTS 16 routine, 2-210  
 TEXT EXTENTS routine, 2-209  
 Text item 16 data structure, 3-58  
 Text item data structure, 3-57  
 Text property data structure, 3-59

TEXT PROPERTY TO STRING LIST routine, 2-211  
 Text Routines  
     DRAW IMAGE STRING, 2-46  
     DRAW IMAGE STRING 16, 2-47  
     DRAW STRING, 2-51  
     DRAW STRING 16, 2-52  
     DRAW TEXT, 2-52  
     DRAW TEXT 16, 2-53  
     QUERY TEXT EXTENTS, 2-144  
     QUERY TEXT EXTENTS 16, 2-145  
     TEXT EXTENTS, 2-209  
     TEXT EXTENTS 16, 2-210  
     TEXT WIDTH, 2-212  
     TEXT WIDTH 16, 2-212  
 Text string (16-bit)  
     drawing, 2-52  
 Text string (8-bit)  
     drawing, 2-51  
 TEXT WIDTH 16 routine, 2-212  
 TEXT WIDTH routine, 2-212  
 Tile  
     best size, 2-139, 2-141  
     changing pixmap, 2-195  
 Tile origin  
     changing in graphics context, 2-196  
 Time coordinate data structure, 3-59  
 TRANSLATE COORDINATES routine, 2-212

## U

---

UNDEFINE CURSOR routine, 2-213  
 Ungrabbing a key, 2-215  
 Ungrabbing a keyboard, 2-215  
 Ungrabbing a pointer, 2-216  
 Ungrabbing server, 2-216  
 UNGRAB BUTTON routine, 2-214  
     See also GRAB BUTTON routine  
 UNGRAB KEYBOARD routine, 2-215  
     See also GRAB KEYBOARD routine  
 UNGRAB KEY routine, 2-215  
     See also GRAB KEY routine  
 UNGRAB POINTER routine, 2-216  
     See also GRAB POINTER routine  
 UNGRAB SERVER routine, 2-216  
     See also GRAB SERVER routine  
 UNINSTALL COLORMAP routine, 2-216  
     See also INSTALL COLORMAP routine  
 Union  
     calculating for region, 2-217  
 UNION RECT WITH REGION routine, 2-217  
 UNION REGION routine, 2-217  
 UNIQUE CONTEXT routine, 2-218  
 UNLOAD FONT routine, 2-218  
 Unmap event data structure, 3-60

- Unmapping all subwindows, 2-218
- Unmapping a window, 2-219
- UNMAP SUBWINDOWS routine, 2-218
- UNMAP WINDOW routine, 2-219
- User environment
  - obtaining defaults, 2-71

## V

---

- Value passing mechanism, 1-5, 1-11
- Vendor
  - identifying, 2-174
- VENDOR RELEASE routine, 2-219
- Visibility event data structure, 3-61
- VISUAL ID FROM VISUAL routine, 2-219
- Visual info data structure, 3-62
- Visual information data structure, 2-91
- Visual type
  - default, 2-34, 2-35

## W

---

- WARP POINTER routine, 2-220
- WHITE PIXEL OF SCREEN routine, 2-221
- WHITE PIXEL routine, 2-220
- WIDTH MM OF SCREEN routine, 2-221
- Width of screen, 2-43, 2-44, 2-221
- WIDTH OF SCREEN routine, 2-221
- Window
  - adding to saveset, 2-15
  - add to save set, 2-2
  - assigning name, 2-206
  - changing and repainting border tile, 2-197
  - changing border width, 2-198
  - changing location and size, 2-127
  - changing parent, 2-152
  - clearing, 2-20
  - configuring, 2-21
  - creating, 2-29, 2-30
  - defining a cursor, 2-35
  - destroying, 2-38
  - getting class hint, 2-70
  - getting size hints, 2-88
  - getting transient for hint, 2-91
  - lowering, 2-124
  - mapping, 2-125
  - mapping and raising, 2-124
  - moving, 2-127
  - obtaining attributes, 2-92
  - obtaining list of children, 2-146
  - obtaining number of children, 2-146
  - obtaining parent, 2-146
  - providing name, 2-57
  - raising, 2-147
  - recommended sizes, 2-83, 2-188
  - recommended zoom sizes, 2-99
  - recommended zoom values, 2-203
  - removing child, 2-151

## Window (cont'd)

- removing from a saveset, 2-15
- resizing, 2-153
- restacking array, 2-154
- setting background, 2-196
- setting background tile, 2-197
- setting border, 2-197
- setting class hint, 2-176
- setting size hints, 2-191
- setting standard properties, 2-192
- setting transient for hint, 2-195
- undefining a cursor, 2-213
- unmapping, 2-219

### Window and Session Manager Routines

- ACTIVATE SCREEN SAVER, 2-1
- ADD HOST, 2-1
- ADD HOSTS, 2-2
- ADD TO SAVE SET, 2-2
- ALLOW EVENTS, 2-7
- AUTO REPEAT OFF, 2-8
- AUTO REPEAT ON, 2-8
- BELL, 2-9
- CHANGE ACTIVE POINTER GRAB, 2-11
- CHANGE KEYBOARD CONTROL, 2-13
- CHANGE KEYBOARD MAPPING, 2-13
- CHANGE POINTER CONTROL, 2-14
- CHANGE SAVE SET, 2-15
- DELETE MODIFIERMAP ENTRY, 2-36
- DISABLE ACCESS CONTROL, 2-38
- DISPLAY KEYCODES, 2-40
- DISPLAY MOTION BUFFER SIZE, 2-40
- ENABLE ACCESS CONTROL, 2-54
- FORCE SCREEN SAVER, 2-62
- FREE MODIFIERMAP, 2-65
- GEOMETRY, 2-67
- GET DEFAULT, 2-71
- GET INPUT FOCUS, 2-80
- GET KEYBOARD CONTROL, 2-80
- GET KEYBOARD MAPPING, 2-81
- GET MODIFIER MAPPING, 2-82
- GET POINTER CONTROL, 2-84
- GET POINTER MAPPING, 2-85
- GET SCREEN SAVER, 2-87
- GRAB BUTTON, 2-100
- GRAB KEY, 2-102
- GRAB KEYBOARD, 2-103
- GRAB POINTER, 2-104
- GRAB SERVER, 2-106
- ICONIFY WINDOW, 2-107
- INSERT MODIFIERMAP ENTRY, 2-108
- INSTALL COLORMAP, 2-109
- KEYCODE TO KEYSYM, 2-110
- KEYSYM TO KEYCODE, 2-111
- KEYSYM TO STRING, 2-111
- KILL CLIENT, 2-112
- LIST HOSTS, 2-117
- LIST INSTALLED COLORMAPS, 2-118
- LOOKUP KEYSYM, 2-123

## Window and Session Manager Routines (cont'd)

- LOOKUP STRING, 2-123
- NEW MODIFIERMAP, 2-128
- PARSE COLOR, 2-131
- PARSE GEOMETRY, 2-132
- QUERY KEYMAP, 2-143
- REBIND KEYSYM, 2-149
- REFRESH KEYBOARD MAPPING, 2-151
- REMOVE FROM SAVE SET, 2-151
- REMOVE HOST, 2-152
- REMOVE HOSTS, 2-152
- REPARENT WINDOW, 2-152
- RESET SCREEN SAVER, 2-153
- RESOURCE MANAGER STRING, 2-154
- SET ACCESS CONTROL, 2-174
- SET CLOSE DOWN MODE, 2-178
- SET INPUT FOCUS, 2-185
- SET MODIFIER MAPPING, 2-187
- SET POINTER MAPPING, 2-189
- SET SCREEN SAVER, 2-190
- STRING TO KEYSYM, 2-207
- UNGRAB BUTTON, 2-214
- UNGRAB KEY, 2-215
- UNGRAB KEYBOARD, 2-215
- UNGRAB POINTER, 2-216
- UNGRAB SERVER, 2-216
- UNINSTALL COLORMAP, 2-216
- WARP POINTER, 2-220
- WM GEOMETRY, 2-223

## Window attributes

- changing, 2-16
- obtaining, 2-92

## Window attributes data structure, 3-63

## Window background

- setting, 2-196

## Window background tile

- setting, 2-197

## Window border

- changing width, 2-198
- setting, 2-197

## Window border tile

- changing, 2-197
- repainting, 2-197

## Window changes data structure, 3-66

## Window coordinates

- transferring, 2-212

## WINDOW EVENT routine, 2-222

## Window manager

- getting hints, 2-96
- regular sizes, 2-83, 2-188
- setting hints, 2-200
- zoom hints, 2-203
- zoom window sizes, 2-99

## Window Routines

- CHANGE WINDOW ATTRIBUTES, 2-16
- CHECK TYPED WINDOW EVENT, 2-17
- CIRCULATE SUBWINDOWS, 2-18
- CIRCULATE SUBWINDOWS DOWN, 2-19

## Window Routines (cont'd)

- CIRCULATE WINDOWS UP, 2-19
- CONFIGURE WINDOWS, 2-21
- CREATE SIMPLE WINDOW, 2-29
- CREATE WINDOW, 2-30
- DESTROY SUBWINDOWS, 2-38
- DESTROY WINDOW, 2-38
- GET GEOMETRY, 2-75
- GET WINDOW ATTRIBUTES, 2-92
- LOWER WINDOW, 2-124
- MAP RAISED, 2-124
- MAP SUBWINDOWS, 2-124
- MAP WINDOW, 2-125
- MOVE RESIZE WINDOW, 2-127
- MOVE WINDOW, 2-127
- QUERY POINTER, 2-143
- QUERY TREE, 2-146
- RAISE WINDOW, 2-147
- RECONFIGURE WM WINDOW, 2-150
- RESIZE WINDOW, 2-153
- RESTACK WINDOWS, 2-154
- SET WINDOW BACKGROUND, 2-196
- SET WINDOW BACKGROUND PIXMAP, 2-197
- SET WINDOW BORDER, 2-197
- SET WINDOW BORDER PIXMAP, 2-197
- SET WINDOW BORDER WIDTH, 2-198
- TRANSLATE COORDINATES, 2-212
- UNMAP SUBWINDOWS, 2-218
- UNMAP WINDOW, 2-219
- WITHDRAW WINDOW, 2-222

## Window selection, 2-22, 2-88, 2-191

## WITHDRAW WINDOW routine, 2-222

## WM GEOMETRY routine, 2-223

## WM hints data structure, 3-66

## WM\_HINTS, 2-200

## WM\_ICON\_SIZES, 2-185

## WM\_NORMAL\_HINTS, 2-188

## WRITE BITMAP FILE routine, 2-223

## Writing bitmap file, 2-223

## X

---

- Xlib Documentation format, 1-1 to 1-5

- Xlib Routine template, 1-1

- XmActivateProtocol routine, 6-7

- XmAddProtocolCallback routine, 6-8

- XmAddProtocols routine, 6-8

- XmAddTabGroup routine, 6-9

- XmAddToPostFromList routine, 6-9

- XmAnyCallbackStruct data structure, 7-3

- XmAnyICCCallbackStruct data structure, 7-3

- XmArrowButtonCallbackStruct data structure, 7-3

- XmCascadeButtonGadgetHighlight routine, 6-9

- XmCascadeButtonHighlight routine, 6-10



XmChangeColor routine, 6-10  
 XmClipboardBeginCopy routine, 6-10  
 XmClipboardCancelCopy routine, 6-11  
 XmClipboardCopyByName routine, 6-12  
 XmClipboardCopy routine, 6-11  
 XmClipboardEndCopy routine, 6-12  
 XmClipboardEndRetrieve routine, 6-13  
 XmClipboardInquireCount routine, 6-13  
 XmClipboardInquireFormat routine, 6-13  
 XmClipboardInquireLength routine, 6-14  
 XmClipboardInquirePendingItems routine, 6-14  
 XmClipboardLock routine, 6-15  
 XmClipboardPendingRec data structure, 7-3  
 XmClipboardRegisterFormat routine, 6-15  
 XmClipboardRetrieve routine, 6-16  
 XmClipboardStartCopy routine, 6-16  
 XmClipboardStartRetrieve routine, 6-17  
 XmClipboardUndoCopy routine, 6-17  
 XmClipboardUnlock routine, 6-17  
 XmClipboardWithdrawFormat routine, 6-18  
 XmCommandAppendValue routine, 6-18  
 XmCommandCallbackStruct data structure, 7-4  
 XmCommandError routine, 6-19  
 XmCommandGetChild routine, 6-19  
 XmCommandSetValue routine, 6-19  
 XmConvertUnits routine, 6-20  
 XmCreateArrowButtonGadget routine, 6-21  
 XmCreateArrowButton routine, 6-20  
 XmCreateBulletinBoardDialog routine, 6-21  
 XmCreateBulletinBoard routine, 6-21  
 XmCreateCascadeButtonGadget routine, 6-22  
 XmCreateCascadeButton routine, 6-22  
 XmCreateCommandDialog routine, 6-23  
 XmCreateCommand routine, 6-23  
 XmCreateDialogShell routine, 6-24  
 XmCreateDragIcon routine, 6-24  
 XmCreateDrawingArea routine, 6-24  
 XmCreateDrawnButton routine, 6-25  
 XmCreateErrorDialog routine, 6-25  
 XmCreateFileSelectionBox routine, 6-26  
 XmCreateFileSelectionDialog routine, 6-26  
 XmCreateFormDialog routine, 6-27  
 XmCreateForm routine, 6-27  
 XmCreateFrame routine, 6-27  
 XmCreateInformationDialog routine, 6-28  
 XmCreateLabelGadget routine, 6-29  
 XmCreateLabel routine, 6-28  
 XmCreateList routine, 6-29  
 XmCreateMainWindow routine, 6-30  
 XmCreateMenuBar routine, 6-30  
 XmCreateMenuShell routine, 6-30  
 XmCreateMessageBox routine, 6-31  
 XmCreateMessageDialog routine, 6-31  
 XmCreateOptionMenu routine, 6-32  
 XmCreatePanedWindow routine, 6-32  
 XmCreatePopupMenu routine, 6-33  
 XmCreatePromptDialog routine, 6-33  
 XmCreatePullDownMenu routine, 6-33  
 XmCreatePushButtonGadget routine, 6-34, 6-42  
 XmCreatePushButton routine, 6-34  
 XmCreateQuestionDialog routine, 6-35  
 XmCreateRadioBox routine, 6-35  
 XmCreateRowColumn routine, 6-36  
 XmCreateScale routine, 6-36  
 XmCreateScrollBar routine, 6-36  
 XmCreateScrolledList routine, 6-37  
 XmCreateScrolledText routine, 6-37  
 XmCreateScrolledWindow routine, 6-38  
 XmCreateSelectionBox routine, 6-38  
 XmCreateSelectionDialog, 6-39  
 XmCreateSeparatorGadget routine, 6-39  
 XmCreateSeparator routine, 6-39  
 XmCreateSimpleCheckBox routine, 6-40  
 XmCreateSimpleMenuBar routine, 6-40  
 XmCreateSimpleOptionMenu routine, 6-41  
 XmCreateSimplePopupMenu routine, 6-41  
 XmCreateSimplePullDownMenu routine, 6-42  
 XmCreateSimpleRadioBox routine, 6-42  
 XmCreateTemplateDialog routine, 6-43  
 XmCreateTextField routine, 6-44  
 XmCreateText routine, 6-43  
 XmCreateToggleButtonGadget routine, 6-45  
 XmCreateToggleButton routine, 6-44  
 XmCreateWarningDialog routine, 6-45  
 XmCreateWorkArea routine, 6-45, 6-46  
 XmCreateWorkingDialog routine, 6-46, 6-47  
 XmCvtCtToXmString routine, 6-47  
 XmCvtStringToUnitType routine, 6-48  
 XmCvtXmStringToCT routine, 6-48  
 XmDeactivateProtocol routine, 6-48  
 XmDestroyPixmap routine, 6-49  
 XmDragCancel routine, 6-49  
 XmDragDropFinishCallbackStruct data structure, 7-4  
 XmDragMotionCallbackStruct data structure, 7-5  
 XmDragProcCallbackStruct data structure, 7-5  
 XmDragStart routine, 6-49  
 XmDrawingAreaCallbackStruct data structure, 7-5  
 XmDrawnButtonCallbackStruct data structure, 7-6  
 XmDropFinishCallbackStruct data structure, 7-6  
 XmDropProcCallbackStruct data structure, 7-4  
 XmDropSiteConfigureStackingOrder routine, 6-50  
 XmDropSiteEndUpdate routine, 6-50  
 XmDropSiteEnterCallbackStruct data structure, 7-6  
 XmDropSiteGetActiveVisuals routine, 6-51  
 XmDropSiteLeaveCallbackStruct data structure, 7-7  
 XmDropSiteQueryStackingOrder routine, 6-51  
 XmDropSiteRegister routine, 6-51

XmDropSiteRetrieve routine, 6-52  
 XmDropSiteStartUpdate routine, 6-52  
 XmDropSiteUnregister routine, 6-53  
 XmDropSiteUpdate routine, 6-53  
 XmDropSiteVisualsRec data structure, 7-7  
 XmDropStartCallbackStruct data structure, 7-8  
 XmDropTransferAdd routine, 6-53  
 XmDropTransferEntryRe data structure, 7-8  
 XmDropTransferStart routine, 6-54  
 XmFileSelectionBoxGetChild routine, 6-54  
 XmFileSelectionCallbackStruct data structure, 7-8  
 XmFileSelectionDoSearch routine, 6-54  
 XmFontListAdd routine, 6-55  
 XmFontListAppendEntry routine, 6-55  
 XmFontListCopy routine, 6-56  
 XmFontListCreate routine, 6-56  
 XmFontListEntryCreate routine, 6-56  
 XmFontListEntryFree routine, 6-57  
 XmFontListEntryGetFont routine, 6-57  
 XmFontListEntryGetTag routine, 6-57  
 XmFontListEntryLoad routine, 6-58  
 XmFontListFreeFontContext routine, 6-59  
 XmFontListFree routine, 6-58  
 XmFontListGetNextFont routine, 6-59  
 XmFontListInitFontContext routine, 6-59  
 XmFontListNextEntry routine, 6-60  
 XmFontListRemoveEntry routine, 6-60  
 XmGetAtomName routine, 6-60  
 XmGetColorCalculation routine, 6-61  
 XmGetColors routine, 6-61  
 XmGetDestination routine, 6-62  
 XmGetDragContext routine, 6-62  
 XmGetFocusWidget routine, 6-62  
 XmGetMenuCursor routine, 6-63  
 XmGetPixmapByDepth routine, 6-63  
 XmGetPixmap routine, 6-63  
 XmGetPostedFromWidget routine, 6-64  
 XmGetTabGroup routine, 6-64  
 XmGetTearOffControl routine, 6-65  
 XmGetVisibility routine, 6-65  
 XmGetXmDisplay routine, 6-65  
 XmGetXmScreen routine, 6-66  
 XmImGetXIM routine, 6-66  
 XmImMbLookupString routine, 6-66  
 XmImRegister routine, 6-67  
 XmImSetFocusValues routine, 6-67  
 XmImSetValues routine, 6-68  
 XmImUnregister routine, 6-68  
 XmImUnsetFocus routine, 6-68  
 XmImVaSetFocusValues routine, 6-69  
 XmImVaSetValues routine, 6-69  
 XmInstallImage routine, 6-69  
 XmInternAtom routine, 6-70  
 XmIsMotifWMRunning routine, 6-70  
 XmIsTraversable routine, 6-70  
 XmListAddItem routine, 6-71  
 XmListAddItems routine, 6-71  
 XmListAddItemsUnselected routine, 6-72  
 XmListAddItemUnselected routine, 6-72  
 XmListDeleteAllItems routine, 6-72  
 XmListDeleteItem routine, 6-73  
 XmListDeleteItemsPos routine, 6-74  
 XmListDeleteItems routine, 6-73  
 XmListDeletePositions routine, 6-74  
 XmListDeletePos routine, 6-74  
 XmListDeselectAllItems routine, 6-75  
 XmListDeselectItem routine, 6-75  
 XmListDeselectPos routine, 6-75  
 XmListGetMatchPos routine, 6-76  
 XmListGetSelectedPos routine, 6-77  
 XmListIGetKbdItemPos routine, 6-76  
 XmListItemExists routine, 6-77  
 XmListItemPos routine, 6-77  
 XmListPosSelected routine, 6-78  
 XmListPosToBounds routine, 6-78  
 XmListReplaceItemsPos routine, 6-79  
 XmListReplaceItemsPosUnselected routine, 6-79  
 XmListReplaceItems routine, 6-79  
 XmListReplaceItemsUnselected routine, 6-80  
 XmListReplacePositions routine, 6-80  
 XmListSelectItem routine, 6-81  
 XmListSelectPos routine, 6-81  
 XmListSetAddMode routine, 6-81  
 XmListSetBottomItem routine, 6-82  
 XmListSetBottomPos routine, 6-82  
 XmListSetHorizPos routine, 6-82  
 XmListSetItem routine, 6-83  
 XmListSetKbdItemPos routine, 6-83  
 XmListSetPos routine, 6-84  
 XmListUpdateSelectedList routine, 6-84  
 XmListYToPos routine, 6-84  
 XmMainWindowSep1 routine, 6-85  
 XmMainWindowSep2 routine, 6-85  
 XmMainWindowSep3 routine, 6-85  
 XmMainWindowSetAreas routine, 6-86  
 XmMapSegmentEncoding routine, 6-86  
 XmMenuPosition routine, 6-86  
 XmMessageBoxGetChild routine, 6-87  
 XmOperationChangedCallbackStruct data structure, 7-9  
 XmOptionButtonGadget routine, 6-87  
 XmOptionLabelGadget routine, 6-88  
 XmProcessTraversal routine, 6-88  
 XmPushButtonCallbackStruct data structure, 7-10  
 XmRegisterConverters routine, 6-88  
 XmRegisterSegmentEncoding routine, 6-88  
 XmRemoveFromPostFromList routine, 6-89  
 XmRemoveProtocolCallback routine, 6-89  
 XmRemoveProtocols routine, 6-90  
 XmRemoveTabGroup routine, 6-90

XmRepTypeAddReverse routine, 6–90  
 XmRepTypeEntryRec data structure, 7–10  
 XmRepTypeGetId routine, 6–91  
 XmRepTypeGetNameList routine, 6–91  
 XmRepTypeGetRecord routine, 6–91  
 XmRepTypeGetRegistered routine, 6–92  
 XmRepTypeInstallTearOffModelConverter routine, 6–92  
 XmRepTypeListRec data structure, 7–10  
 XmRepTypeRegister routine, 6–92  
 XmRepTypeValidValue routine, 6–92  
 XmResolveAllPartOffsets routine, 6–93  
 XmResolvePartOffsets routine, 6–93  
 XmRowColumnCallbackStruct data structure, 7–11  
 XmScaleCallbackStruct data structure, 7–11  
 XmScaleGetValue routine, 6–94  
 XmScaleSetValue routine, 6–94  
 XmScrollBarCallbackStruct data structure, 7–11  
 XmScrollBarGetValues routine, 6–94  
 XmScrollBarSetValues routine, 6–95  
 XmScrolledWindowSetAreas routine, 6–96  
 XmScrollVisible routine, 6–95  
 XmSecondaryResourceDataRec data structure, 7–12  
 XmSelectionBoxCallbackStruct data structure, 7–12  
 XmSelectionBoxGetChild routine, 6–96  
 XmSetColorCalculation routine, 6–96  
 XmSetFontUnit routine, 6–97  
 XmSetFontUnits routine, 6–97  
 XmSetMenuCursor routine, 6–98  
 XmSetProtocolHooks routine, 6–98  
 XmStringBaseline routine, 6–98  
 XmStringByteCompare routine, 6–99  
 XmStringCompare routine, 6–99  
 XmStringConcat routine, 6–100  
 XmStringCopy routine, 6–100  
 XmStringCreateLocalized routine, 6–101  
 XmStringCreateLtoR routine, 6–101  
 XmStringCreate routine, 6–100  
 XmStringCreateSimple routine, 6–101  
 XmStringDirectionCreate routine, 6–102  
 XmStringDrawImage routine, 6–103  
 XmStringDraw routine, 6–102  
 XmStringDrawUnderline routine, 6–103  
 XmStringEmpty routine, 6–104  
 XmStringExtent routine, 6–104  
 XmStringFreeContext routine, 6–105  
 XmStringFree routine, 6–105  
 XmStringGetLtoR routine, 6–105  
 XmStringGetComponent routine, 6–106  
 XmStringGetNextSegment routine, 6–106  
 XmStringHasSubstring routine, 6–107  
 XmStringHeight routine, 6–107  
 XmStringInitContext routine, 6–107  
 XmStringLength routine, 6–108  
 XmStringLineCount routine, 6–108  
 XmStringNConcat routine, 6–108  
 XmStringNCopy routine, 6–109  
 XmStringPeekNextComponent routine, 6–109  
 XmStringSegmentCreate routine, 6–110  
 XmStringSeparatorCreate routine, 6–110  
 XmStringWidth routine, 6–110  
 XmTargetsAreCompatible routine, 6–111  
 XmTextBlockRec data structure, 7–12  
 XmTextBlockRecWcs data structure, 7–13  
 XmTextClearSelection routine, 6–111  
 XmTextCopy routine, 6–112  
 XmTextCut routine, 6–112  
 XmTextDisableRedisplay routine, 6–112  
 XmTextEnableRedisplay routine, 6–113  
 XmTextFieldClearSelection routine, 6–113  
 XmTextFieldCopy routine, 6–113  
 XmTextFieldCut routine, 6–114  
 XmTextFieldGetBaseline routine, 6–114  
 XmTextFieldGetCursorPosition routine, 6–114  
 XmTextFieldGetEditable routine, 6–115  
 XmTextFieldGetInsertionPosition routine, 6–115  
 XmTextFieldGetLastPosition routine, 6–115  
 XmTextFieldGetMaxLength routine, 6–116  
 XmTextFieldGetSelectionPosition routine, 6–116  
 XmTextFieldGetSelection routine, 6–116  
 XmTextFieldGetSelectionWcs routine, 6–117  
 XmTextFieldGetString routine, 6–117  
 XmTextFieldGetStringWcs routine, 6–117  
 XmTextFieldGetSubstring routine, 6–118  
 XmTextFieldGetSubstringWcs routine, 6–118  
 XmTextFieldInsert routine, 6–119  
 XmTextFieldInsertWcs routine, 6–119  
 XmTextFieldPaste routine, 6–119  
 XmTextFieldPosToXY routine, 6–120  
 XmTextFieldRemove routine, 6–120  
 XmTextFieldReplace routine, 6–121  
 XmTextFieldReplaceWcs routine, 6–121  
 XmTextFieldSetAddMode routine, 6–121  
 XmTextFieldSetCursorPosition routine, 6–122  
 XmTextFieldSetEditable routine, 6–122  
 XmTextFieldSetHighlight routine, 6–123  
 XmTextFieldSetInsertionPosition routine, 6–123  
 XmTextFieldSetMaxLength routine, 6–123  
 XmTextFieldSetSelection routine, 6–124  
 XmTextFieldSetString routine, 6–124  
 XmTextFieldShowPosition routine, 6–124  
 XmTextFieldXYToPos routine, 6–125  
 XmTextFindString routine, 6–125  
 XmTextFindStringWcs routine, 6–126  
 XmTextGetBaseline routine, 6–126  
 XmTextGetCursorPosition routine, 6–126  
 XmTextGetEditable routine, 6–127  
 XmTextGetInsertionPosition routine, 6–127  
 XmTextGetLastPosition routine, 6–128



XmTextGetMaxLength routine, 6-128  
 XmTextGetSelectionPosition routine, 6-129  
 XmTextGetSelection routine, 6-128  
 XmTextGetSelectionWcs routine, 6-129  
 XmTextGetSource routine, 6-129  
 XmTextGetString routine, 6-130  
 XmTextGetStringWcs routine, 6-130  
 XmTextGetSubstring routine, 6-130  
 XmTextGetSubstringWcs routine, 6-131  
 XmTextGetTopCharacter routine, 6-131  
 XmTextInsert routine, 6-132  
 XmTextInsertWcs routine, 6-132  
 XmTextPaste routine, 6-132  
 XmTextPosToXY routine, 6-133  
 XmTextRemove routine, 6-133  
 XmTextReplace routine, 6-134  
 XmTextReplaceWcs routine, 6-134  
 XmTextScroll routine, 6-134  
 XmTextSetAddMode routine, 6-135  
 XmTextSetCursorPosition routine, 6-135  
 XmTextSetEditable routine, 6-136  
 XmTextSetHighlight routine, 6-136  
 XmTextSetInsertionPosition routine, 6-136  
 XmTextSetMaxLength routine, 6-137  
 XmTextSetSelection routine, 6-137  
 XmTextSetSource routine, 6-137  
 XmTextSetString routine, 6-138  
 XmTextSetStringWcs routine, 6-138  
 XmTextSetTopCharacter routine, 6-139  
 XmTextShowPosition routine, 6-139  
 XmTextVerifyCallbackStruct data structure, 7-13  
 XmTextVerifyCallbackStructWcs data structure, 7-13  
 XmTextXYToPos routine, 6-139  
 XmToggleButtonCallbackStruct data structure, 7-14  
 XmToggleButtonGadgetGetState routine, 6-140  
 XmToggleButtonGadgetSetState routine, 6-140  
 XmToggleButtonGetState routine, 6-140  
 XmToggleButtonSetState routine, 6-141  
 XmTopLevelEnterCallbackStruct data structure, 7-14  
 XmTopLevelLeaveCallbackStruct data structure, 7-14  
 XmTrackingEvent routine, 6-141  
 XmTrackingLocate routine, 6-142  
 XmTranslateKey routine, 6-142  
 XmTraverseObscuredCallbackStru data structure, 7-15  
 XmUninstallImage routine, 6-142  
 XmUpdateDisplay routine, 6-143  
 XmVaCreateSimpleCheckBox routine, 6-143  
 XmVaCreateSimpleMenuBar routine, 6-144  
 XmVaCreateSimpleOptionMenu routine, 6-144  
 XmVaCreateSimplePopupMenu routine, 6-144  
 XmVaCreateSimplePullDownMenu routine, 6-145  
 XmVaCreateSimpleRadioBox routine, 6-145  
 XmWidgetGetBaselines routine, 6-146  
 XmWidgetGetDisplayRect routine, 6-146  
 XOR REGION routine, 2-224  
 XtActionsRec data structure, 5-1  
 XtAddActions routine, 4-1  
 XtAddCallback routine, 4-1  
 XtAddCallbacks routine, 4-2  
 XtAddConverter routine, 4-2  
 XtAddEventHandler routine, 4-2  
 XtAddExposuretoRegion routine, 4-3  
 XtAddGrab routine, 4-3  
 XtAddInput routine, 4-4  
 XtAddRawEventHandler routine, 4-4  
 XtAddTimeout routine, 4-5  
 XtAddWorkProc routine, 4-5  
 XtAllocateGC routine, 4-5  
 XtAppAddActionHook routine, 4-6  
 XtAppAddActions routine, 4-6  
 XtAppAddConverter routine, 4-7  
 XtAppAddInput routine, 4-7  
 XtAppAddTimeout routine, 4-8  
 XtAppAddWorkProc routine, 4-8  
 XtAppCreateShell routine, 4-8  
 XtAppErrorMsg routine, 4-9  
 XtAppError routine, 4-9  
 XtAppGetErrorDatabase routine, 4-10  
 XtAppGetErrorDatabaseText routine, 4-10  
 XtAppGetSelectionTimeout routine, 4-11  
 XtAppInitialize routine, 4-11  
 XtAppMainLoop routine, 4-12  
 XtAppNextEvent routine, 4-12  
 XtAppPeekEvent routine, 4-12  
 XtAppPending routine, 4-13  
 XtAppProcessEvent routine, 4-13  
 XtAppReleaseCacheRefs routine, 4-13  
 XtAppSetErrorHandler routine, 4-14  
 XtAppSetErrorMsgHandler routine, 4-14  
 XtAppSetFallbackResources routine, 4-14  
 XtAppSetSelectionTimeout routine, 4-15  
 XtAppSetTypeConverter routine, 4-15  
 XtAppSetWarningHandler routine, 4-16  
 XtAppSetWarningMsgHandler routine, 4-16  
 XtAppWarningMsg routine, 4-16  
 XtAppWarning routine, 4-16  
 XtAugmentTranslations routine, 4-17  
 XtBuildEventMask routine, 4-17  
 XtCallAcceptFocus routine, 4-18  
 XtCallActionProc routine, 4-18  
 XtCallbackExclusive routine, 4-19  
 XtCallbackNone routine, 4-19  
 XtCallbackNonexclusive routine, 4-19  
 XtCallbackPopdown routine, 4-20  
 XtCallbackRec data structure, 5-1  
 XtCallbackReleaseCacheRefList routine, 4-21  
 XtCallbackReleaseCacheRef routine, 4-20

XtCallCallbackList routine, 4-21  
 XtCallCallbacks routine, 4-21  
 XtCallConverter routine, 4-22  
 XtCalloc routine, 4-22  
 XtClass routine, 4-23  
 XtCloseDisplay routine, 4-23  
 XtConvertAndStore routine, 4-24  
 XtConvertArgRec data structure, 5-2  
 XtConvertCase routine, 4-24  
 XtConvert routine, 4-23  
 XtCreateApplicationContext routine, 4-25  
 XtCreateApplicationShell routine, 4-25  
 XtCreateManagedWidget routine, 4-25  
 XtCreatePopupShell routine, 4-26  
 XtCreateWidget routine, 4-26  
 XtCvtColorToPixel routine, 4-27  
 XtCvtIntToBoolean routine, 4-28  
 XtCvtIntToBool routine, 4-27  
 XtCvtIntToColor routine, 4-28  
 XtCvtIntToFloat routine, 4-29  
 XtCvtIntToFont routine, 4-29  
 XtCvtIntToPixel routine, 4-30  
 XtCvtIntToPixmap routine, 4-30  
 XtCvtIntToShort routine, 4-31  
 XtCvtIntToUnsignedChar routine, 4-31  
 XtCvtStringToAcceleratorTable routine, 4-32  
 XtCvtStringToAtom routine, 4-32  
 XtCvtStringToBoolean routine, 4-33  
 XtCvtStringToBool routine, 4-33  
 XtCvtStringToCursor routine, 4-34  
 XtCvtStringToDimension routine, 4-34  
 XtCvtStringToDisplay routine, 4-35  
 XtCvtStringToFile routine, 4-35  
 XtCvtStringToFloat routine, 4-36  
 XtCvtStringToFont routine, 4-36  
 XtCvtStringToFontSet routine, 4-37  
 XtCvtStringToFontStruct routine, 4-37  
 XtCvtStringToInitialState routine, 4-38  
 XtCvtStringToInt routine, 4-38  
 XtCvtStringToPixel routine, 4-39  
 XtCvtStringToShort routine, 4-39  
 XtCvtStringToTranslationTable routine, 4-40  
 XtCvtStringToUnsignedChar routine, 4-40  
 XtCvtStringToVisual routine, 4-41  
 XtDatabase routine, 4-41  
 XtDestroyApplicationContext routine, 4-41  
 XtDestroyGC routine, 4-42  
 XtDestroyWidget routine, 4-42  
 XtDirectConvert routine, 4-43  
 XtDisownSelection routine, 4-43  
 XtDispatchEvent routine, 4-43  
 XtDisplayInitialize routine, 4-44  
 XtDisplayOfObject routine, 4-45  
 XtDisplay routine, 4-44  
 XtDisplayStringConvWarning routine, 4-45  
 XtDisplayToApplicationContext routine, 4-45  
 XtErrorMsg routine, 4-46  
 XtError routine, 4-46  
 XtFindFile routine, 4-46  
 XtFree routine, 4-47  
 XtGetActionKeysym routine, 4-48  
 XtGetActionList routine, 4-47  
 XtGetApplicationNameAndClass routine, 4-48  
 XtGetApplicationResources routine, 4-48  
 XtGetConstraintResourceList routine, 4-49  
 XtGetErrorDatabase routine, 4-49  
 XtGetErrorDatabaseText routine, 4-50  
 XtGetGC routine, 4-50  
 XtGetKeysymTable routine, 4-51  
 XtGetMultiClickTime routine, 4-51  
 XtGetResourceList routine, 4-51  
 XtGetSelectionRequest routine, 4-52  
 XtGetSelectionTimeout routine, 4-52  
 XtGetSelectionValueIncremental routine, 4-53  
 XtGetSelectionValue routine, 4-53  
 XtGetSelectionValuesIncremental routine, 4-54  
 XtGetSelectionValues routine, 4-54  
 XtGetSubresources routine, 4-55  
 XtGetSubvalues routine, 4-55  
 XtGetValues routine, 4-56  
 XtGrabButton routine, 4-56  
 XtGrabKeyboard routine, 4-57  
 XtGrabKey routine, 4-57  
 XtGrabPointer routine, 4-58  
 XtHasCallbacks routine, 4-58  
 XtI18nContextRec data structure, 5-2  
 XtInitialize routine, 4-59  
 XtInitializeWidgetClass routine, 4-59  
 XtInsertEventHandler routine, 4-59  
 XtInsertRawEventHandler routine, 4-60  
 XtInstallAccelerators routine, 4-60  
 XtInstallAllAccelerators routine, 4-61  
 XtISensitive routine, 4-62  
 XtIsManaged routine, 4-61  
 XtIsObject routine, 4-61  
 XtIsRealized routine, 4-62  
 XtIsSubclass routine, 4-62  
 XtKeysymToKeycodeList routine, 4-63  
 XtLastTimestampProcessed routine, 4-63  
 XtMainLoop routine, 4-63  
 XtMakeGeometryRequest routine, 4-64  
 XtMakeResizeRequest routine, 4-64  
 XtMalloc routine, 4-64  
 XtManageChildren routine, 4-65  
 XtManageChild routine, 4-65  
 XtMenuPopupAction routine, 4-65  
 XtMergeArgLists routine, 4-66  
 XtName routine, 4-66  
 XtNameToWidget routine, 4-67  
 XtNextEvent routine, 4-67  
 XtOpenDisplay routine, 4-67  
 XtOverrideTranslations routine, 4-68

XtOwnSelectionIncremental routine, 4-69  
XtOwnSelection routine, 4-68  
XtParent routine, 4-69  
XtParseAcceleratorTable routine, 4-70  
XtParseTranslationTable routine, 4-70  
XtPeekEvent routine, 4-70  
XtPending routine, 4-71  
XtPopdownIdRec data structure, 5-2  
XtPopdown routine, 4-71  
XtPopup routine, 4-72  
XtPopupSpringLoaded routine, 4-72  
XtProcessEvent routine, 4-72  
XtQueryGeometry routine, 4-73  
XtRealizeWidget routine, 4-73  
XtRealloc routine, 4-73  
XtRegisterCaseConverter routine, 4-74  
XtRegisterGrabAction routine, 4-74  
XtReleaseGC routine, 4-75  
XtRemoveActionHook routine, 4-75  
XtRemoveAllCallbacks routine, 4-75  
XtRemoveCallback routine, 4-76  
XtRemoveCallbacks routine, 4-76  
XtRemoveEventHandler routine, 4-76  
XtRemoveGrab routine, 4-77  
XtRemoveInput routine, 4-77  
XtRemoveRawEventHandler routine, 4-78  
XtRemoveTimeOut routine, 4-78  
XtRemoveWorkProc routine, 4-78  
XtResolvePathname routine, 4-79  
XtResource data structure, 5-2  
XtScreenDatabase routine, 4-80  
XtScreenOfObject routine, 4-80  
XtScreen routine, 4-79  
XtSetErrorHandler routine, 4-80  
XtSetErrorMsgHandler routine, 4-81  
XtSetKeyboardFocus routine, 4-81  
XtSetKeyTranslator routine, 4-81  
XtSetLanguageProc routine, 4-82  
XtSetMappedWhenManaged routine, 4-82  
XtSetMultiClickTime routine, 4-82  
XtSetSelectionTimeout routine, 4-83  
XtSetSensitive routine, 4-83  
XtSetSubvalues routine, 4-84  
XtSetTypeConverter routine, 4-84  
XtSetValues routine, 4-85  
XtSetWarningHandler routine, 4-85  
XtSetWarningMsgHandler routine, 4-85  
XtSetWMCColormapWindows routine, 4-86  
XtStringConversionWarning routine, 4-86  
XtSubstitutionRec data structure, 5-3  
XtSuperclass routine, 4-86  
XtToolkitInitialize routine, 4-87  
XtTranslateCoords routine, 4-87  
XtTranslateKeycode routine, 4-88  
XtTranslateKey routine, 4-87  
XtUngrabButton routine, 4-88  
XtUngrabKeyboard routine, 4-89  
XtUngrabKey routine, 4-89  
XtUngrabPointer routine, 4-89  
XtUninstallTranslations routine, 4-90  
XtUnmanageChildren routine, 4-90  
XtUnmanageChild routine, 4-90  
XtUnrealizeWidget routine, 4-91  
XtVaAppCreateShell routine, 4-91  
XtVaAppInitialize routine, 4-91  
XtVaCreateArgsList routine, 4-92  
XtVaCreateManagedWidget routine, 4-92  
XtVaCreatePopupShell, 4-93  
XtVaCreateWidget routine, 4-93  
XtVaGetApplicationResources routine, 4-94  
XtVaGetSubresources routine, 4-94  
XtVaGetSubvalues routine, 4-95  
XtVaGetValues routine, 4-95  
XtVaSetSubvalues routine, 4-96  
XtVaSetValues routine, 4-96  
XtWarningMsg routine, 4-97  
XtWarning routine, 4-96  
XtWidgetGeometry data structure, 5-3  
XtWidgetToApplicationContext routine, 4-97  
XtWindowOfObject routine, 4-98  
XtWindow routine, 4-98  
XtWindowToWidget routine, 4-98

