Xmu and Xmuu Library API Reference

X Version 11, Release 7.7

Copyright © 1989 X Consortium

! "

$

%

#&' ()#*+,' - ,(.-/'/ + - *-#&(0# *+,,+1#2 () +12
3-1/ '4 ,' (, -5 6-'/ -1760/-18 90# 1(# 6-5-#'/ #(#&'
*+,,+1#-' () 5',7&+1#+9-6-#2)-#1')(, + +,#-706+,
0, (' +1/ 1(1-1),-18'5'1#% -1 1('.'1# &+66 #&' 4
7(1 (,#-05 9' 6-+96')(, +12 76+-5 /+5+8' (, (#&', 6-+9-6-
#2 *&'#&', -1 +1 +7#-(1 () 7(1#,+7# #(,# (, (#&',*- '
+,- -18),(5 (0# () (, -1 7(11'7#-(1 *-#& #&' ()#*+,' (,
#&' 0 ' (, (#&', /'+6-18 -1 #&' ()#*+,'%

': 4 7

$

; 4 7 %

™ < # (8 %

Abstract

/ = <%

Table of Contents

- %%%%

Introduction

4 6
$ $ % #
A + * %

4) BC D%>%E 4>>,C%F
+ * 4 4 # <
- 4 $ 4 %

< $ **xmu** **xmuu**
"

```
cc -c xapplication.c `pkg-config --cflags xmu`
cc -o xapplication xapplication.o `pkg-config --libs xmu`
```

Atom Functions

#
G X11/Xmu/Atoms.hH < 4 %

XA_ATOM_PAIR(*d*);

XA_CHARACTER_POSITION(*d*);

XA_CLASS(*d*);

XA_CLIENT_WINDOW(*d*);

XA_CLIPBOARD(*d*);

XA_COMPOUND_TEXT(*d*);

XA_DECNET_ADDRESS(*d*);

XA_DELETE(*d*);

XA_FILENAME(*d*);

XA_HOSTNAME(*d*);

XA_IP_ADDRESS(*d*);

XA_LENGTH(*d*);

XA_LIST_LENGTH(*d*);

XA_NAME(*d*);

XA_NET_ADDRESS(*d*);

XA_NULL(*d*);

XA_OWNER_OS(*d*);

XA_SPAN(*d*);

XA_TARGETS(*d*);

XA_TEXT(*d*);

XA_TIMESTAMP(*d*);

XA_USER(*d*);

XA_UTF8_STRING(*d*);

< + %#
$ XA_%#
+ $ A A
$ %

AtomPtr **XmuMakeAtom**(*name*);

name

; A ! + +
$ % XmuInternAtom + $
%

char ***XmuNameOfAtom**(*atom_ptr*);

atom_ptr +

+ %

Atom **XmuInternAtom**(*dpy*, *atom_ptr*);

dpy 4 $

atom_ptr +

+ + %# +
A A $ %

char ***XmuGetAtomName**(*dpy*, *atom*);

dpy 4 $

atom

+ %#
A A $ %

```
void XmuInternStrings(dpy, names, count, atoms);
```

dpy 4 $

names

count

atoms + $

$ + $ %#
A A $ % #
%

Error Handler Functions

G X11/
Xmu/Error.hH < 4 4 %

```
int XmuPrintDefaultErrorMessage(dpy, event, fp);
```

dpy 4 $

event

fp

A $ 4 =
% - ; $:
E% #
%

```
int XmuSimpleErrorHandler(dpy, errorp);
```

dpy 4 $

errorp

9 * XQueryTree XGetWin-
dowAttributes 9 / XGetGeometryI E
% (
$: E : %

System Utility Functions

G X11/
Xmu/SysUtil.hH < 4 4 %

```
int XmuGetHostname(buf, maxlen);
```

buf

maxlen *buf*

buf
%#
gethostname uname%

int **XmuSnprintf**(*str*, *size*, ...);

str

size ; *str*

$ snprintf
$ %- $
snprintf
0 :JB %

Window Utility Functions

G X11/
Xmu/WinUtil.hH %# XmuClientWindow < 4
4 %# A <
4 %

Screen ***XmuScreenOfWindow**(*dpy*, *w*);

dpy 4 $

w

%

Window **XmuClientWindow**(*dpy*, *win*);

dpy 4 $

win

#
*5K #+#' %-
%

Bool **XmuUpdateMapHints**(*dpy*, *w*, *hints*);

dpy 4 $

win

hints 1066

; 0
0 ;
XSetWMNormalHints # % - 1066
< XGetWMNormalHints
I) %

Cursor Utility Functions

G X11/
Xmu/CurUtil.hH < 4 4 %

int **XmuCursorNameToIndex**(*name*);

name

< :
% # $ XC_ :
+ :9 4 %

Graphics Functions

G X11/
Xmu/Drawing.hH < 4 %

void **XmuDrawRoundedRectangle**(*dpy*, *draw*, *gc*, *x*, *y*, *w*, *h*, *ew*, *eh*);

dpy 4 $

draw

gc 87

x :

y

w

h

ew

eh

:
$; :
I
% # 87 %

void **XmuFillRoundedRectangle**(*dpy*, *draw*, *gc*, *x*, *y*, *w*, *h*, *ew*, *eh*);

dpy 4 $

draw

gc 87

x :

y

w

h

ew

eh

:
$; :
I
%# 87 %

XmuDrawLogo(*dpy*, *drawable*, *gcFore*, *gcBack*, *x*, *y*, *width*, *height*);

dpy 4 $

drawable

gcFore 87

gcBack < 87

x :

y

width

height

4 *) T> # 4 6 %
: $: %#
) 9 <%

Pixmap **XmuCreateStippledPixmap**(*screen*, *fore*, *back*, *depth*);

screen :

fore : $

back < : $

depth :

: : :
%# : A
: %# : XmuReleaseStippledPixmap
%

void **XmuReleaseStippledPixmap**(*screen*, *pixmap*);

screen :

pixmap :

: XmuCreateStippledPixmap%

int **XmuReadBitmapData**(*fstream*, *width*, *height*, *datap*, *x_hot*, *y_hot*);

fstream

width

height

datap

x_hot :

y_hot

#
XCreateBitmapFromData%
$ $
XReadBitmapFile%

int **XmuReadBitmapDataFromFile**(*filename*, *width*, *height*, *datap*, *x_hot*, *y_hot*);

fstream

width

height

datap

x_hot :

y_hot

#

XCreateBitmapFromData%

$ $

XReadBitmapFile%

Pixmap **XmuLocateBitmapFile**(*screen*, *name*, *srcname*, *srcnamelen*, *widthp*, *heightp*, *xhotp*, *yhotp*);

screen :

name

srcname

srcnamelen

width

height

xhotp :

yhotp

XReadBitmapFile

% # $

bitmapFilePath 9) % -

$ 9-#5+ /-,

L/usr/include/X11/bitmapsL% - *srcnamelen* ; *srcname*

1066 *srcname*% # ;

%

Pixmap **XmuCreatePixmapFromBitmap**(*dpy*, *d*, *bitmap*, *width*, *height*, *depth*, *fore*, *back*);

dpy 4 $

screen :

bitmap

width :

height :

depth :

fore : $

back < : $

: XCopyPlane
: <
: $ % # : %

Selection Functions

G X11/ Xmu/StdSel.hH < 4 %

Boolean **XmuConvertStandardSelection**(*w*, *time*, *selection*, *target*, *type*, *value*, *length*, *format*);

w

time

selection

target

type $ $

value $ $

length $ $

format ;

$ " 76+ 76-'1#K*-1/(* /'71'#K+//,' &(#1+5' - K+//,' 1+5' (*1',K(#+,8'# #-5' #+5 0 ',% - # $) %

Type Converter Functions

G X11/ Xmu/Converters.hH < 4 %

void **XmuCvtFunctionToCallback**(*args*, *num_args*, *fromVal*, *toVal*);

args

num_args

fromVal $

toVal $ $

$ < < 1066 % # $ = 7 - ; "

```
XtAddConverter(XtRCallProc, XtRCallback, XmuCvtFunctionToCallback, NULL, 0);
```

```
void XmuCvtStringToBackingStore(args, num_args, fromVal, toVal);
```

args

num_args 7 $ E

fromVal $

toVal $ $

$ < G X11/X.hH%
L 0 L $ 1 0 L 5 L $ *
5 L L $ + %# L L $ $
+ M* 5 M1 0 %# %#
$ = 7 - ; "

```
XtAddConverter(XtRString, XtRBackingStore, XmuCvtStringToBackingStore, NULL, 0);
```

```
void XmuCvtStringToBitmap(args, num_args, fromVal, toVal);
```

args

num_args $ >

fromVal $

toVal $ $

:
%# %)
XmuLocateBitmapFile N>E@% #
$ = 7 - ; "

```
static XtConvertArgRec screenConvertArg[] = {
  {XtBaseOffset, (XtPointer)XtOffset(Widget, core.screen), sizeof(Screen *)}
};

XtAddConverter(XtRString, XtRBitmap, XmuCvtStringToBitmap,
   screenConvertArg, XtNumber(screenConvertArg));
```

```
Boolean XmuCvtStringToColorCursor(dpy, args, num_args, fromVal, toVal,
data);
```

dpy $

args A $

num_args A $ D

fromVal $

toVal $ $

data

$ 7 <
: $ % #
$ "XC_" :
+ :9 4 5 :
L)(1# :NN O :OL
XmuLocateBitmapFile N>E@% # $
7 - ; "

```
static XtConvertArgRec colorCursorConvertArgs[] = {
  {XtWidgetBaseOffset, (XtPointer) XtOffsetOf(WidgetRec, core.screen),
   sizeof(Screen *)},
  {XtResourceString, (XtPointer) XtNpointerColor, sizeof(Pixel)},
  {XtResourceString, (XtPointer) XtNpointerColorBackground, sizeof(Pixel)},
  {XtWidgetBaseOffset, (XtPointer) XtOffsetOf(WidgetRec, core.colormap),
   sizeof(Colormap)}
};


XtSetTypeConverter(XtRString, XtRColorCursor, XmuCvtStringToColorCursor,
   colorCursorConvertArgs, XtNumber(colorCursorConvertArgs),
                XtCacheByDisplay, NULL);
```

; 41 7 41 7 9 <
< % #
= , ; . $ $ <
<

%

void **XmuCvtStringToCursor**(*args*, *num_args*, *fromVal*, *toVal*);

args A $

num_args A $ >

fromVal $

toVal $ $

$ 7 %
$ "XC_" :
+ :9 4 5 :

L)(1# : NN O :OL
XmuLocateBitmapFile N>E@% # $
= 7 - ; "

```
static XtConvertArgRec screenConvertArg[] = {
  {XtBaseOffset, (XtPointer)XtOffsetOf(WidgetRec, core.screen), sizeof(Screen *)}
};

XtAddConverter(XtRString, XtRCursor, XmuCvtStringToCursor,
   screenConvertArg, XtNumber(screenConvertArg));
```

```
void XmuCvtStringToGravity(args, num_args, fromVal, toVal);
```

args

num_args 7 $ E

fromVal $

toVal $ $

$ 4 8 $ $ % # L
L 1066 $ $) 8 $ L1 * 8 $ L $
1 * 8 $ L1 8 $ L L L $ 1 8 $
L1 ' 8 $ L $ 1 ' 8 $ L* L L L
$ * 8 $ L7 8 $ L $ 7 8 $ L' 8 $ L
L L $ ' 8 $ L * 8 $ L $ * 8 $
L 8 $ L L L $ 8 $ L ' 8 $ L $
' 8 $ L 8 $ L $ 8 $ L0 8 $ L
$ 0 8 $ % # % #
$ = ; "

```
XtAddConverter(XtRString, XtRGravity, XmuCvtStringToGravity, NULL, 0);
```

```
void XmuCvtStringToJustify(args, num_args, fromVal, toVal);
```

args

num_args

fromVal $

toVal $ $

$ 4 P $ % # L L
$ 4 P 6 L L $ 4 P 7 L L $
4 P , % # % # $
= 7 - ; "

XtAddConverter(XtRString, XtRJustify, XmuCvtStringToJustify, NULL, 0);

void **XmuCvtStringToLong**(*args*, *num_args*, *fromVal*, *toVal*);

args

num_args 7 E

fromVal $

toVal $ $

$ %-
sscanf LQ L% # $
= 7 - ; "

XtAddConverter(XtRString, XtRLong, XmuCvtStringToLong, NULL, 0);

void **XmuCvtStringToOrientation**(*args*, *num_args*, *fromVal*, *toVal*);

args

num_args

fromVal $

toVal $ $

$ 4($ %#
L ; L $ 4 & ; L$ L $ 4 .
%# %# $
= 7 - ; "

XtAddConverter(XtRString, XtROrientation, XmuCvtStringToOrientation, NULL, 0);

Boolean **XmuCvtStringToShapeStyle**(*dpy*, *args*, *num_args*, *fromVal*, *toVal*, *data*);

dpy $

args

num_args

fromVal $ $

toVal $ $

data

$ %# L L
$ 4 , L$ L $ 4 ($ L L $
4 ' L , L $ 4 , ,
%# %# $
= 7 - ; "

```
XtSetTypeConverter(XtRString, XtRShapeStyle, XmuCvtStringToShapeStyle,
        NULL, 0, XtCacheNone, NULL);
```

Boolean **XmuReshapeWidget**(*w*, *shape_style*, *corner_width*, *corner_height*);

w

shape_style

corner_width

corner_height

:
$ K 4 ,
4 ($ 4 ' 4 , ,
$ %# :
%- K K
; : Xmu-
FillRoundedRectangle NF0 K K
%# %

void **XmuCvtStringToWidget**(*args*, *num_args*, *fromVal*, *toVal*);

args *

num_args >

fromVal $

toVal $ $

$
%1 $ <
$ I ; $ %#

%-

%# %# $
= 7 - ; "

```
static XtConvertArgRec parentCvtArg[] = {
  {XtBaseOffset, (XtPointer)XtOffset(Widget, core.parent), sizeof(Widget)},
};

XtAddConverter(XtRString, XtRWidget, XmuCvtStringToWidget,
   parentCvtArg, XtNumber(parentCvtArg));
```

Boolean **XmuNewCvtStringToWidget**(*dpy*, *args*, *num_args*, *fromVal*, *toVal*, *data*);

dpy $

args *

num_args 7 $ >

fromVal $

toVal $ $

data

$ XmuCvtStringToWidget N>CO :
$
% 5 $

$ %# $ =
; "

```
static XtConvertArgRec parentCvtArg[] = {
  {XtWidgetBaseOffset, (XtPointer)XtOffsetOf(WidgetRec, core.parent),
   sizeof(Widget)}
};

XtSetTypeConverter(XtRString, XtRWidget, XmuNewCvtStringToWidget,
     parentCvtArg, XtNumber(parentCvtArg), XtCacheNone, NULL);
```

Character Set Functions

G X11/
Xmu/CharSet.hH < 4 4 %

= <
4>>I $ $
%

void **XmuCopyISOLatin1Lowered**(*dst*, *src*);

dst

src

#
6 > % #
-(BBRJ >%

1 < strcpy ; *dst*
; *src*%

void **XmuNCopyISOLatin1Lowered**(*dst*, *src*, *size*);

dst

src

size :

size - 1
src *dst* 6 >
% # -(BBRJ >%

void **XmuCopyISOLatin1Uppered**(*dst*, *src*);

dst

src

#
6 > % #
-(BBRJ >%

1 < strcpy ; *dst*
; *src*%

void **XmuNCopyISOLatin1Uppered**(*dst*, *src*, *size*);

dst

src

size :

size - 1
src *dst* 6 >
% # -(BBRJ >%

```
int XmuCompareISOLatin1(first, second);
```

first

second

6 >
A E
: A % #
- (BBRJ >%

Keyboard Event Translation Functions

G X11/
Xmu/Lookup.hH % < 4 %

= <
4>>I XmbLookupString $
$ %

```
int XmuLookupLatin1(event, buffer, nbytes, keysym, status);
```

event < $

buffer

nbytes

keysym 3 1

status

XLookupString :
%

```
int XmuLookupLatin2(event, buffer, nbytes, keysym, status);
```

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $
6 @ - (BBRJ @ + 7-- %

int **XmuLookupLatin3**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $
6 S - (BBRJ S + 7-- %

int **XmuLookupLatin4**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $
6 D - (BBRJ D + 7-- %

int **XmuLookupKana**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $
6 > - (BBRJ > + 7-- 8
6 $ E >@F 3 < 8 , $ >@B
@RR $ P- 4@E> >JFC%

int **XmuLookupJISX0201**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $
P- 4E@E> >JFC + 7-- %

int **XmuLookupArabic**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $ 6
+ - (BBRJ C + 7-- %

int **XmuLookupCyrillic**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $ 6
7 - (BBRJ R + 7-- %

int **XmuLookupGreek**(*event*, *buffer*, *nbytes*, *keysym*, *status*);

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $ 6
8 < - (BBRJ F + 7-- %

```
int XmuLookupHebrew(event, buffer, nbytes, keysym, status);
```

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $ 6
& -(BBRJ B +7-- %

```
int XmuLookupAPL(event, buffer, nbytes, keysym, status);
```

event < $

buffer

nbytes

keysym 3 1

status

XLookupString : < $
+6 %

Compound Text Functions

7 #:
$ %/
G X11/Xmu/Xct.hH < 4 %

#
% #
0#)BK #,-18 : $ $ %

+7 #: "

```
typedef unsigned char *XctString;
```

```
XctData XctCreate(string, length, flags);
```

string 7 #:

length

flags

4 / 7 #: %
%
"

4 # 7E 7> 86 8, % *
A 4 $ Xct-
NextItem 4 7E 4 7> 4
8 4 8, % 7E 7>
%

4 $ ': # 7 #:
$
< A
4 ': XctNex-
tItem% - 7 #:
$:
< A
4 ' %

4 + 7E': # 7 #:
$
< 7E
7E
4 $ ': XctNextItem% -
< 7E
4 $ ': %

4 + 7>': # 7 #:
$
< 7>
7>
4 $ ': XctNextItem% -
< 7>
4 $ ': %

4 & / # ;
4 & ; XctNextItem% -

% # $
$ 4 0
4 6 #, 4 , #6 %

4) # XctFree 7 #:
XctCreate% -
%

4 5 8,#86 # XctNextItem
8, 86

8, " 89@S>@%>JBE > P- 4E@EB%>JBS >
3 7RCE>%>JBF >%

```
void XctReset(data);
```

data 7 #:

4 / 7 #:
%

```
XctResult XctNextItem(data);
```

data 7 #:

: 7 #: %# $
< %#
: 4 / %4 ,
$ "

4 : 7E 86 8, 7>
%

4 7E 7E %

4 86 86 %

4 7> 7> %

4 8, 8, %

4 ': : %

4 ': < : A %

4 & ; ; %
4 /
%

4 ' (#: # 7 #: %

4 ' I
%

$ 4 / "

```
XctString  item;  /* the action item */
unsigned  item_length; /* length of item in bytes */
int   char_size; /* the number of bytes per character in
  * item, with zero meaning variable */
char  *encoding; /* the XLFD encoding name for item */
XctHDirection horizontal; /* the direction of item */
```

```
    unsigned  horz_depth; /* the current direction nesting depth */
    char  *GL;  /* the "{I} F" string for the current GL */
    char  *GL_encoding; /* the XLFD encoding name for the current GL */
    int   GL_set_size; /* 94 or 96 */
    int   GL_char_size; /* the number of bytes per GL character */
    char  *GR;  /* the "{I} F" string for the current GR */
    char  *GR_encoding; /* the XLFD encoding name for the current GR */
    int   GR_set_size; /* 94 or 96 */
    int   GR_char_size; /* number of bytes per GR character */
    char  *GLGR_encoding; /* the XLFD encoding name for the current
      *  GL+GR, if known */
```

void **XctFree**(*data*);

data 7 #:

4 / %

CloseDisplay Hook Functions

G X11/ Xmu/CloseHook.hH < 4 %

CloseHook **XmuAddCloseDisplayHook**(*dpy*, *(*func)*, *arg*);

dpy 4 $

func

arg *func*

< $ % *
$ $ "

(*func)(dpy, arg)

$ $
$ $ %

1066 <
A $ < <%

Bool **XmuRemoveCloseDisplayHook**(*dpy*, *handle*, *(*func)*, *arg*);

dpy 4 $

handle < 1066

func <

arg

< XmuAddCloseDisplayHook%
- 1066 < $
% - 1066 <
$ % , # < $
) %

Bool **XmuLookupCloseDisplayHook**(*dpy*, *handle*, (**func*), *arg*);

dpy 4 $

handle < 1066

func <

arg

< % - 1066
< < % -
1066 < < %,
< :) %

Display Queue Functions

#
G X11/Xmu/DisplayQue.hH < 4 % DisplayQue.h
"

```
typedef int (*XmuCloseDisplayQueueProc)(XmuDisplayQueue *queue,
                                        XmuDisplayQueueEntry *entry);

typedef int (*XmuFreeDisplayQueueProc)(XmuDisplayQueue *queue);

typedef struct _XmuDisplayQueueEntry {
 struct _XmuDisplayQueueEntry *prev, *next;
 Display *display;
 CloseHook closehook;
 XPointer data;
} XmuDisplayQueueEntry;

typedef struct _XmuDisplayQueue {
 int nentries;
 XmuDisplayQueueEntry *head, *tail;
 XmuCloseDisplayQueueProc closefunc;
 XmuFreeDisplayQueueProc freefunc;
 XPointer data;
} XmuDisplayQueue;
```

XmuDisplayQueue ***XmuDQCreate**(*closefunc*, *freefunc*, *data*);

closefunc

freefunc

data $

4 / ? !
A % # A
XmuAddDisplay% # $
A < % * $
A XCloseDisplay *closefunc* 1066
A = XmuDisplayQueueEntry "

(*closefunc)(queue, entry)

freefunc 1066 $ A
"

(*freefunc)(queue)

A XmuDQDe-
stroy%

XmuDisplayQueueEntry ***XmuDQAddDisplay***(*q*, *dpy*, *data*);

q A

dpy

data $

A % - A
1066 % # $
A A = <% #
$ A I XmuDQLookupDisplay
A %

XmuDisplayQueueEntry ***XmuDQLookupDisplay***(*q*, *dpy*);

q A

dpy <

A 1066
A %

XmuDQNDisplays(*q*);

A %

Bool **XmuDQRemoveDisplay**(*q*, *dpy*);

q A

dpy $

$ A % 1 <
% - A)
# %

Bool **XmuDQDestroy**(*q*, *docallbacks*);

q A

docallbacks

A % -
< # A = < 1066
A $ XCloseDisplay %

Toolkit Convenience Functions

G X11/
Xmu/Iniater.hH < 4 %

void **XmuAddInitializer**(*(*func)*, *data*);

func

data $

$ < XmuCallInitial-
izers $: % #
: "

(*func)(app_con, data)

void **XmuCallInitializers**(*app_con*);

app_con : ;

$ XmuAd-
dInitializer :
XmuCallInitializers%(%

Standard Colormap Functions

G X11/
Xmu/StdCmap.hH < 4 %

Status **XmuAllStandardColormaps**(*dpy*);

dpy 4 $

$ $ $
$ XmuAllStandardColormaps%

#
$ % - E
; % -
%

#
%

#
= % #
: ,89K/')+06#K5+
%

" ,89K9' #K5+ ,89K,'/K5+ ,89K8,"1K5+
,89K960'K5+ ,89K/')+06#K5+ ,89K8,+2K5+% #
$ C %

+ $ % +
$ $ $
% 1 $
$ $ $ % # $
$
% 9
$
$ < %

" $
$
" / 7 I 7 I # 7 8 I
7 8 %

%) :
> E % & $
> % - E %

XmuVisualStandardColormaps N@JO
$ %

Status **XmuVisualStandardColormaps**(*dpy*, *screen*, *visualid*, *depth*, *replace*, *retain*);

dpy 4 $

screen

visualid $

depth $

replace

retain

$ $ $
XmuVisualStandardColormaps%

$
$ %- # $ $ %- #
$ %#
E ; % (
$ $ # %

1 $ % #
<
$; %) /
7 7 " ,89K/')+06#K5+ ,89K9' #K5+ ,89K,'/K5+
,89K8,"1K5+ ,89K960'K5+ ,89K8,+2K5+%) # 7
7 " ,89K9' #K5+%) 8 8 " ,89K8,+2K5+%

Status **XmuLookupStandardColormap**(*dpy*, *screen*, *visualid*, *depth*, *property*, *replace*, *retain*);

dpy 4 $

screen

visualid $

depth $

property

replace

retain

:
: XmuLookupStandardColormap%

8 $ $
$
%

- # $ %-
#
$ % & $:

I A
%

E ; % + A
$
% + : A
$ A ,89K9' #K5+
; >C%

Status **XmuGetColormapAllocation**(*vinfo*, *property*, *red_max*, *green_max*, *blue_max*);

vinfo $ $

property

red_max : $

green_max : $

blue_max : $

#
XmuGetColormapAllocation%

XmuGetColormapAllocation E ; %-
$ %

XStandardColormap ***XmuStandardColormap**(*dpy*, *screen*, *visualid*, *depth*, *property*, *cmap*, *red_max*, *green_max*, *blue_max*);

dpy 4 $

screen

visualid $

depth $

property

cmap -/ 1

red_max

green_max

blue_max

XmuStandardColormap%

$ $ $
$: $ $
% 0 XStandardColormap

% 0 1066% -

I %

, I =
%

Status **XmuCreateColormap**(*dpy*, *colormap*);

dpy

colormap

4 7
XmuCreateColormap%

E ; % # K :
% ,
% 1 < $ I <%

+ :
: : $
$ % #
% #
* 9 < %

7 $
$ $ %

void **XmuDeleteStandardColormap**(*dpy*, *screen*, *property*);

dpy 4 $

screen

property

$ XmuDeleteStandardColormap%
$
%

Widget Description Functions

#
% # <
%

G X11/Xmu/
WidgetNode.hH % #
< 4 %
"

void **XmuWnInitializeNodes**(*node_array*, *num_nodes*);

node_array

num_nodes

$

void **XmuWnFetchResources**(*node*, *toplevel*, *top_node*);

node %

toplevel *node*
: % # $
XtAppInitialize%

top_node *node*

%

' % #
"

int **XmuWnCountOwnedResources**(*node*, *owner_node*, *constraints*);

node : %

owner_node *node*
%

constraints
%

#
%

XmuWidgetNode ***XmuWnNameToNode**(*node_list*, *num_nodes*, *name*);

node_list

num_nodes

name

* 1 $
% - 1066%

Participation in the Editres Protocol

#
+ G X11/Xmu/Editres.hH <
4 %

4 +

$ < E < $
1066"

```
XtAddEventHandler(shell, (EventMask) 0, True, _XEditResCheckMessages, NULL);
```

Index

D D D D R R

A

+ D

B

< >@
) >E

C

@J
7 >@ >S

E

SS

P

< S

S

: >C

W

>@
*5K #+#' C

X

4>> 4 + % S
4>> 4 7 % >F
4>> 4 7 & <% @R
4>> 4 7 $ % >>
4>> 4 7 0 % F
4>> 4 / ? % @C
4>> 4 / % F
4>> 4 ' % SS
4>> 4 ' % R
4>> 4 - % @B
4>> 4 6 < % >J
4>> 4 7 % @J
4>> 4 % >>
4>> 4 0 % R
4>> 4 * 1 % S@
4>> 4 * 0 % C
4>> 4 4 % @@

4 7 @@
4 / @D
4 1 : - @D
4 , @D
48 *51 & C
4 + 7 / & < @R
4 + - ; @B
4 + 7 @J
4 7 - ; @B
4 7 * C
4 7 - (6 > >B
4 7 $ >>
4 7 - (6 >6 >F
4 7 - (6 >0 >B
4 7 7 S@
4 7 :) 9 >E
4 7 : B
4 7 1 #- : F
4 7$) #7 < >>
4 7$ #9 < >@
4 7$ #9 >@
4 7$ #7 7 >@
4 7$ #7 >S
4 7$ #8 $ >D
4 7$ #P >D
4 7$ #6 >R
4 7$ #(>R
4 7$ # >R
4 7$ #* >C
4 / 7 S@
4 / ? @C
4 /?+ / @F
4 /?7 @C
4 /?/ @B
4 /?6 < / @F
4 /?1/ @F
4 /?, $ / @B
4 / 6 B
4 / , , F
4) , , F
4 8 7 + S>
4 8 & R
4 6 9) >E
4 6 < + 6 @@
4 6 < + @>
4 6 < 7 / & < @C
4 6 < 7 @>
4 6 < 8 < @>
4 6 < & @@
4 6 < P- 4E@E> @E
4 6 < 3 @E
4 6 < 6 > >J
4 6 < 6 @ >J
4 6 < 6 S >J

4 6 < 6 D @E
4 6 < 7 SE
4 17 - (6 >6 >B
4 17 - (6 >0 >B
4 1 7$ #* >F
4 , 9 / J
4 , 9 /)) J
4 , : J
4 , $ 7 / & < @R
4 , * >C
4 (* C
4 ' & R
4 C
4 7 S>
4 0 5 & C
4 . 7 @J
4 * 1 #1 SS
4 *51 & C
4 8 $ >D
4 P >D
4 (>R