NAME

XtMalloc, XtCalloc, XtRealloc, XtReallocArray, XtFree, XtNew, XtNewString - memory management
functions

SYNTAX

#include <X11/Intrinsic.h>

char *XtMalloc(Cardinal *size*);

char *XtCalloc(Cardinal *num*, Cardinal *size*);

char *XtRealloc(char *ptr*, Cardinal *size*);

void *XtReallocArray(void *ptr*, Cardinal *num*, Cardinal *size*);

void XtFree(char *ptr*);

type *XtNew(*type*);

String XtNewString(String *string*);

Cardinal XtAsprintf(char **new_string*, const char *format*, ...);

ARGUMENTS

num   Specifies the number of array elements.

ptr   Specifies a pointer to the old storage or to the block of storage that is to be freed.

size   Specifies the size of an array element (in bytes) or the number of bytes desired.

string   Specifies a previously declared string.

type   Specifies a previously declared data type.

new_string   Specifies a pointer to write a newly allocated string to.

format   Specifies a formatting string as defined by sprintf(3c)

DESCRIPTION

The **XtMalloc** functions returns a pointer to a block of storage of at least the specified size bytes. If

there is insufficient memory to allocate the new block, **XtMalloc** calls **XtErrorMsg**.

The **XtCalloc** function allocates space for the specified number of array elements of the specified size and initializes the space to zero. If there is insufficient memory to allocate the new block, **XtCalloc** calls **XtErrorMsg**.

The **XtRealloc** and **XtReallocArray** functions change the size of a block of storage (possibly moving it). Then, they copy the old contents (or as much as will fit) into the new block and free the old block. If there is insufficient memory to allocate the new block, or the calculations for the size of the new block would cause an integer overflow, these functions call **XtErrorMsg**. If ptr is NULL, these functions allocate the new storage without copying the old contents; that is, they simply call **XtMalloc**.

The **XtFree** function returns storage and allows it to be reused. If ptr is NULL, **XtFree** returns immediately.

XtNew returns a pointer to the allocated storage. If there is insufficient memory to allocate the new block, **XtNew** calls **XtErrorMsg**. **XtNew** is a convenience macro that calls **XtMalloc** with the following arguments specified:

    ((type *) XtMalloc((unsigned) sizeof(type))

XtNewString returns a pointer to a new string which is a duplicate of *string*. If there is insufficient memory to allocate the new block, or the argument is NULL **XtNewString** returns NULL. The memory can be freed with **XtFree**.

The **XtAsprintf** function allocates space for a string large enough to hold the string specified by the sprintf(3c) format pattern when used with the remaining arguments, and fills it with the formatted results. The address of the allocated string is placed into the pointer passed as ret. The length of the string (not including the terminating null byte) is returned. If there is insufficient memory to allocate the new block, **XtAsprintf** calls **XtErrorMsg**.

SEE ALSO
    X Toolkit Intrinsics - C Language Interface
    Xlib - C Language X Interface