

NAME

XStringListToTextProperty, XTextPropertyToStringList, XFreeStringList, XTextProperty - convert string lists and text property structure

SYNTAX

```
Status XStringListToTextProperty(char **list, int count, XTextProperty *text_prop_return);
```

```
Status XTextPropertyToStringList(XTextProperty *text_prop, char ***list_return, int *count_return);
```

```
void XFreeStringList(char **list);
```

ARGUMENTS

- count* Specifies the number of strings.
- count_return* Returns the number of strings.
- list* Specifies the list of strings to be freed.
- list* Specifies a list of null-terminated character strings.
- list_return* Returns a list of null-terminated character strings.
- text_prop* Specifies the **XTextProperty** structure to be used.
- text_prop_return*
Returns the **XTextProperty** structure.

DESCRIPTION

The **XStringListToTextProperty** function sets the specified **XTextProperty** to be of type STRING (format 8) with a value representing the concatenation of the specified list of null-separated character strings. An extra null byte (which is not included in the `nitems` member) is stored at the end of the value field of `text_prop_return`. The strings are assumed (without verification) to be in the STRING encoding. If insufficient memory is available for the new value string, **XStringListToTextProperty** does not set any fields in the **XTextProperty** structure and returns a zero status. Otherwise, it returns a nonzero status. To free the storage for the value field, use **XFree**.

The **XTextPropertyToStringList** function returns a list of strings representing the null-separated elements of the specified **XTextProperty** structure. The data in `text_prop` must be of type STRING and format 8. Multiple elements of the property (for example, the strings in a disjoint text selection) are separated by NULL (encoding 0). The contents of the property are not null-terminated. If insufficient

memory is available for the list and its elements, **XTextPropertyToStringList** sets no return values and returns a zero status. Otherwise, it returns a nonzero status. To free the storage for the list and its contents, use **XFreeStringList**.

The **XFreeStringList** function releases memory allocated by **XmbTextPropertyToTextList**, **Xutf8TextPropertyToTextList** and **XTextPropertyToStringList** and the missing charset list allocated by **XCreateFontSet**.

STRUCTURES

The **XTextProperty** structure contains:

```
typedef struct {
    unsigned char *value; /* property data */
    Atom encoding; /* type of property */
    int format; /* 8, 16, or 32 */
    unsigned long nitems; /* number of items in value */
} XTextProperty;
```

SEE ALSO

XAllocClassHint(3), **XAllocIconSize(3)**, **XAllocSizeHints(3)**, **XAllocWMHints(3)**, **XFree(3)**, **XSetCommand(3)**, **XSetTransientForHint(3)**, **XSetTextProperty(3)**, **XSetWMClientMachine(3)**, **XSetWMColormapWindows(3)**, **XSetWMIconName(3)**, **XSetWMName(3)**, **XSetWMProperties(3)**, **XSetWMProtocols(3)**

Xlib - C Language X Interface