XSetFontPath, XGetFontPath, XFreeFontPath – set, get, or free the font search path

XSetFontPath(*display***,** *directories***,** *ndirs***)**
Display ***display;*
char ****directories;*
int *ndirs*;

char **XGetFontPath(*display*, *npaths_return*)
Display **display*;
int **npaths_return*;

XFreeFontPath(*list*)
char ***list*;

directories	Specifies the directory path used to look for a font. Setting the path to the empty list restores the default path defined for the X server.
display	Specifies the connection to the X server.
list	Specifies the array of strings you want to free.
ndirs	Specifies the number of directories in the path.
npaths_return	Returns the number of strings in the font path array.

The XSetFontPath function defines the directory search path for font lookup. There is only one search path per X server, not one per client. The encoding and interpretation of the strings are implementation-dependent, but typically they specify directories or font servers to be searched in the order listed. An X server is permitted to cache font information internally; for example, it might cache an entire font from a file and not check on subsequent opens of that font to see if the underlying font file has changed. However, when the font path is changed, the X server is guaranteed to flush all cached information about fonts for which there currently are no explicit resource IDs allocated. The meaning of an error from this request is implementation-dependent.

XSetFontPath can generate a **BadValue** error.

The **XGetFontPath** function allocates and returns an array of strings containing the search path. The contents of these strings are implementation-dependent and are not intended to be interpreted by client applications. When it is no longer needed, the data in the font path should be freed by using **XFreeFontPath**.

The **XFreeFontPath** function frees the data allocated by **XGetFontPath**.

BadValue Some numeric value falls outside the range of values accepted by the request. Unless a specific 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.

XListFont(3X11), XLoadFonts(3X11)
Xlib – C Language X Interface