

NAME

XtResolvePathname – search for a file using standard substitution

SYNTAX

String XtResolvePathname(Display **display*, String *type*, String *filename*, String *suffix*, String *path*, Substitution *substitutions*, Cardinal *num_substitutions*, XtFilePredicate *predicate*);

ARGUMENTS

display Specifies the display to use to find the language for language substitutions.

type

filename

suffix Specify values to substitute into the path.

path Specifies the list of file specifications, or NULL.

substitutions Specifies a list of additional substitutions to make into the path, or NULL.

num_substitutions Specifies the number of entries in *substitutions*.

predicate Specifies a procedure called to judge each potential file name, or NULL.

DESCRIPTION

The substitutions specified by **XtResolvePathname** are determined from the value of the language string retrieved by **XtDisplayInitialize** for the specified display. To set the language for all applications specify “*xnLanguage: *lang*” in the resource database. The format and content of the language string are implementation-defined. One suggested syntax is to compose the language string of three parts; a “language part”, a “territory part” and a “codeset part”. The manner in which this composition is accomplished is implementation-defined and the Intrinsics make no interpretation of the parts other than to use them in substitutions as described below.

XtResolvePathname calls **XtFindFile** with the following substitutions in addition to any passed by the caller and returns the value returned by **XtFindFile**:

%N The value of the *filename* parameter, or the application’s class name if *filename* is NULL.

%T The value of the *type* parameter.

%S The value of the *suffix* parameter.

%L The language string associated with the specified display.

%l The language part of the display’s language string.

%t The territory part of the display’s language string.

%c The codeset part of the display’s language string.

%C The customization string retrieved from the resource database associated with *display*.

%D The value of the implementation-specific default path.

If a path is passed to **XtResolvePathname**, it will be passed along to **XtFindFile**. If the *path* argument is NULL, the value of the **XFILESEARCHPATH** environment variable will be passed to **XtFindFile**. If **XFILESEARCHPATH** is not defined, an implementation-specific default path will be used which contains at least 6 entries. These entries must contain the following substitutions:

1. %C, %N, %S, %T, %L or %C, %N, %S, %T, %l, %t, %c
2. %C, %N, %S, %T, %l
3. %C, %N, %S, %T
4. %N, %S, %T, %L or %N, %S, %T, %l, %t, %c
5. %N, %S, %T, %l
6. %N, %S, %T

The order of these six entries within the path must be as given above. The order and use of substitutions within a given entry is implementation dependent. If the path begins with a colon, it will be preceded by

`%N%S`. If the path includes two adjacent colons, `%N%S` will be inserted between them.

The *type* parameter is intended to be a category of files, usually being translated into a directory in the path-name. Possible values might include “app-defaults”, “help”, and “bitmap”.

The *suffix* parameter is intended to be appended to the file name. Possible values might include “.txt”, “.dat”, and “.bm”.

A suggested value for the default path on POSIX-based systems is

```
/usr/lib/X11/%L/%T/%N%C%S:/usr/lib/X11/%L/%T/%N%C%S:\
/usr/lib/X11/%T/%N%C%S:/usr/lib/X11/%L/%T/%N%S:\
/usr/lib/X11/%L/%T/%N%S:/usr/lib/X11/%T/%N%S
```

Using this example, if the user has specified a language, it will be used as a subdirectory of `/usr/lib/X11` that will be searched for other files. If the desired file is not found there, the lookup will be tried again using just the language part of the specification. If the file is not there, it will be looked for in `/usr/lib/X11`. The *type* parameter is used as a subdirectory of the language directory or of `/usr/lib/X11`, and *suffix* is appended to the file name.

The `%D` substitution allows the addition of path elements to the implementation-specific default path, typically to allow additional directories to be searched without preventing resources in the system directories from being found. For example, a user installing resource files under a directory called “ourdir” might set **XFILESEARCHPATH** to

```
%D:ourdir/%T/%N%C:ourdir/%T/%N
```

The customization string is obtained by querying the resource database currently associated with the display (the database returned by **XrmGetDatabase**) for the resource *application_name*.customization, class *application_class*.Customization where *application_name* and *application_class* are the values returned by **XtGetApplicationNameAndClass**. If no value is specified in the database, the empty string is used.

It is the responsibility of the caller to free the returned string using **XtFree** when it is no longer needed.

SEE ALSO

X Toolkit Intrinsics – C Language Interface
Xlib – C Language X Interface