

**NAME**

XtFindFile – search for a file using substitutions in the path list

**SYNTAX**

```
#include <X11/Intrinsic.h>
```

```
char * XtFindFile(const char *path, Substitution substitutions, Cardinal num_substitutions, XtFilePredicate  
                predicate);
```

**ARGUMENTS**

<i>path</i>	Specifies a path of file names, including substitution characters.
<i>substitutions</i>	Specifies a list of substitutions to make into a path.
<i>num_substitutions</i>	Specifies the number of substitutions passed in.
<i>predicate</i>	Specifies a procedure to call to judge a potential file name, or NULL.

**DESCRIPTION**

The *path* parameter specifies a string that consists of a series of potential file names delimited by colons. Within each name, the percent character specifies a string substitution selected by the following character. The character sequence “%:” specifies an embedded colon that is not a delimiter; the sequence is replaced by a single colon. The character sequence “%%” specifies a percent character that does not introduce a substitution; the sequence is replaced by a single percent character. If a percent character is followed by any other character, **XtFindFile** looks through the specified *substitutions* for that character in the *match* field and if found replaces the percent and match characters with the string in the corresponding *substitution* field. A *substitution* field entry of NULL is equivalent to a pointer to an empty string. If the operating system does not interpret multiple embedded name separators in the path (i.e., “/” in POSIX) the same way as a single separator, **XtFindFile** will collapse multiple separators into a single one after performing all string substitutions. Except for collapsing embedded separators, the contents of the string substitutions are not interpreted by **XtFindFile** and may therefore contain any operating-system-dependent characters, including additional name separators. Each resulting string is passed to the predicate procedure until a string is found for which the procedure returns **True**; this string is the return value for **XtFindFile**. If no string yields a **True** return from the predicate, **XtFindFile** returns NULL.

If the *predicate* parameter is NULL, an internal procedure that checks if the file exists, is readable, and is not a directory will be 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*