NAME

intro – introduces the PEXlib graphics library

DESCRIPTION

PEXlib is a programmer's interface to the PEX protocol. PEX is an extension to the X window system for supporting three-dimensional graphics.

The application can query the existence of the PEX server extension by calling **XQueryExtension** using the name, **PEX_NAME_STRING,** defined in PEXlib include files.

To begin using PEXlib, the application must initialize PEXlib by calling **PEXInitialize**(3) before calling any PEXlib function which takes a display argument. Functions which do not take a display argument may be called prior to calling **PEXInitialize**(3)**. PEXInitialize**(3) must be called for each display connection the application will use. Failure to call **PEXInitialize**(3) will result in undefined behavior for functions which take a display argument. Termination of PEXlib occurs implicitly at either program end or **XCloseDisplay.**

There are four other functions that should be called during initialization to determine what features and parameters are supported by each particular PEX server implementation. These functions are **PEXGetExtensionInfo**(3)**, PEXGetImpDepConstants**(3)**, PEXGetEnumTypeInfo**(3)**,** and **PEXMatchRenderingTargets**(3)**. PEXGetExtensionInfo**(3) returns information about the PEX server extension such as protocol version, vendor string, release number and subset support. **PEXGetImpDepConstants**(3) returns information about supported implementation-dependent constants. **PEXGetEnumTypeInfo**(3) returns information about supported enumerated types. **PEXMatchRenderingTargets**(3) returns information about supported drawable types.

Some PEXlib functions allocate memory on behalf of the application. Memory allocation often occurs in the case of inquiry functions where the size of data to be returned is unknown. Applications must use **XFree** to deallocate memory that has been allocated for them by PEXlib, unless a specific function is provided to free memory allocated for particular PEXlib objects. Where necessary, the specific function for deallocation is noted in the description of the function which allocates the memory. Memory allocated by PEXlib is considered to be "owned" by PEXlib; the application must not modify pointers to memory returned by the library nor attempt to free such memory except by the specified interfaces. Data structure fields other than pointers to memory can be changed without consequence.

PEXlib is designed as an extension to Xlib using the same transport and error mechanisms as Xlib. This includes all requests, replies, events and errors. This means synchronization (e.g. **XSync** and **XSynchronize**) will have the same effect on PEXlib requests as they do on Xlib requests. PEX events also are treated in the same way as X events (e.g. use **XNextEvent**).

PEXlib functions usually do not check for invalid parameters. Applications should be careful to pass correct data to PEXlib. Incorrect values sent to the PEX server extension will cause an error event to be returned to the client. These errors are asynchronous, and so are difficult to trace to a particular Xlib or PEXlib call unless using **XSync** or **XSynchronize .** Note these functions should be used with prudence, since they slow client programs significantly.

By default, when an error event is returned by the server, Xlib prints a message and the application does not continue. The application can register its own error handler by calling **XSetErrorHandler .**

The floating point format used by PEXlib is the format native to the architecture of the client machine (i.e. the machine on which the application executes). All floating point values are expected to be in that native format, with a few exceptions for functions which allow the application to specify the floating point format. Those functions which allow the application to specify a floating point format deal primarily with application data that is already in protocol format, or data that the application wants to be encoded into protocol format for its own use.

When the server to which PEXlib is sending data does not support the floating point format native to PEXlib, PEXlib will choose an appropriate format which is supported by the server and convert into that format when formatting protocol to send to that server. In cases where PEXlib must choose a non-native floating point format, the application can determine the format chosen by calling This is useful with those functions dealing with protocol formatted data to help avoid unnecessary format conversions when sending the data to the PEX server.

The PEXlib specification currently defines function bindings and include files for the C language only. The specification is intended to work for both ANSI C and Kernighan and Ritchie (K&R) C compilers.

The **< X11/PEX5/PEX.h >** and **< X11/PEX5/PEXlib.h >** include files contain constant definitions, macros, and data structures used by PEXlib. **< X11/PEX5/PEXlib.h >** automatically includes **< X11/PEX5/PEX.h >.** Some data structures have fields named "reserved". These fields are necessary to align the data correctly for efficient transfer to the protocol transport buffer. Any values assigned to these fields will be ignored.

The library is specified in the compilation command as "-lPEX5".