NAME

PEXBeginPickAll – Begin Renderer Pick All

SYNTAX

void PEXBeginPickAll(Display **display*, Drawable *drawable*, PEXRenderer *renderer*, long *structure_id*, int *method*, int *send_event*, int *max_hits*, int *pick_device_type*, PEXPickRecord **pick_record*)

PARAMETERS

display	A pointer to a display structure returned by a successful **XOpenDisplay** call.
drawable	The resource identifier of a drawable.
renderer	The resource identifier of the renderer.
structure_id	A value to be used as the structure identifier for the root of the structure network.
method	The pick all method (**PEXPickAllAll** or **PEXPickAllVisible**).
send_event	**TRUE** or **FALSE** — specifying whether the server should send an event when the maximum number of hits is reached.
max_hits	The maximum number of hits to be returned.
pick_device_type	The pick device type (**PEXPickDeviceDCHitBox** or **PEXPickDeviceNPCHitVolume**).
pick_record	A pointer to the pick data record.

RETURNS

None

DESCRIPTION

This functions starts an immediate-mode pick, setting the renderer's renderer state to **PEXPicking.** When the renderer state is **PEXPicking,** primitives are hit tested instead of converted to pixels. All picked primitives are recorded until reaching the maximum hits specified is reached. Additional picked primitives will not be recorded. Once the the maximum number of hits is reached, subsequent primitives may be ignored.

The supported pick device types are inquirable via **PEXGetEnumTypeInfo**(3)**.** The specified structure identifier will be inserted as the first structure component in the returned pick path(s).

If the *send_event* flag is **True,** and the pick *method* is **PEXPickAllAll,** then a **PEXMaxHitsReached** event is sent from the server to the client whenever the maximum number of hits is reached by the server, if the event is supported (see **PEXGetImpDepConstants**(3)). Upon receiving the event, the application should stop sending primitives and process the recorded hits. If the pick *method* is **PEXPickAllVisible,** a complete set of primitives must be sent to the server before determining which primitives are picked.

If the specified drawable does not have the same root and depth as the drawable used to create the renderer, or, if the specified drawable is not one of the supported drawables returned by **PEXMatchRenderingTargets**(3)**,** a match error is generated. If the renderer state is set to **PEXRendering** or **PEXPicking** when this function is called, then the operation in progress is aborted, the **PEXBeginPickAll** function is completed, and a

BadPEXRendererState error returned.

All functions which process output commands or manipulate attributes (i.e. all output command functions, **PEXBeginStructure**(3)**, PEXEndStructure**(3)**, PEXRenderElements**(3)**,** and **PEXAccumulateState**(3)) can be called when the renderer state is **PEXPicking.** They will have the same semantics except that primitives are hit tested instead of converted to pixels.

DATA STRUCTURES

```
typedef  XID     PEXRenderer;

typedef union {
        PEXPDNPCHitVolume       volume;
        PEXPDDCHitBox                   box;
        PEXPickDataRecord       data;
} PEXPickRecord;

typedef PEXNPCSubVolume     PEXPDNPCHitVolume;

typedef struct {
        PEXCoord        min;
        PEXCoord        max;
} PEXNPCSubVolume;

typedef struct {
        float     x;
        float     y;
        float     z;
} PEXCoord;

typedef struct {
        PEXDeviceCoord2D     position;
        float                distance;
} PEXPDDCHitBox;

typedef struct {
        short    x;
        short    y;
} PEXDeviceCoord2D;

typedef struct {
        unsigned short          length;         /* number of bytes in record */
        char                    *record;
} PEXPickDataRecord;
```

```
typedef struct {
        int                     type;
        unsigned long           serial;         /* # of last request processed by server */
        Bool                    send_event;     /* true if this came from a SendEvent reques
        Display                 *display;       /* Display the event was read from */
        PEXRenderer             renderer;
} PEXMaxHitsReachedEvent;
```

ERRORS

BadAlloc
The server failed to allocate resources necessary to complete request.

BadDrawable
The specified drawable resource identifier is invalid.

BadMatch
The specified drawable is unsupported, or the specified renderer resource was not created with a compatible drawable.

BadPEXRenderer
The specified renderer resource identifier is invalid.

BadPEXRendererState
The specified renderer was in an invalid state.

BadValue
The pick record contains invalid data, or the pick device type is invalid.

SEE ALSO

PEXEndPickAll(3)
PEXPickAll(3)
PEXGetImpDepConstants(3)