NAME PEXAnnotationText – 3D Annotation Text Primitive

SYNTAX void PEXAnnotationText(Display **display*, XID *resource_id*, PEXOCRequestType *req_type*, PEXCoord **origin*, PEXCoord **offset*, int *length*, char **string*)

PARAMETERS

display	A pointer to a display structure returned by a successful **XOpenDisplay** call.
resource_id	The resource identifier of the renderer or structure.
req_type	The request type for the output command (**PEXOCRender, PEXOCStore, PEXOCRenderSingle** or **PEXOCStoreSingle**).
origin	The origin of the text string.
offset	The relative position of the text string from the origin.
length	The number of bytes in the text string.
string	A pointer to the text string.

RETURNS None

DESCRIPTION This function creates an annotation text output primitive.

The first character set in the text font will be used.

The *origin* defines, in modeling coordinates, the position of the first character in the text *string.* The *offset* specifies an offset from the *origin* in normalized projection coordinates. The point where the text *string* is placed is called the annotation point. The annotation point is computed by adding the value of *offset* to the transformed *origin* point. The z-component of the annotation point specifies the x-y plane in normalized projection coordinate space on which the annotation text *string* is placed.

Depending on the text attribute source flag values, the text color, text precision, character expansion, character spacing, and text font index attributes are either obtained directly from the current text attribute values or from the entry in the text bundle specified by the current text bundle index attribute. The annotation text height, annotation text path, annotation text alignment, annotation text up vector, and annotation text style are also used to render the text *string.*

The annotation text string’s color is affected only by the depth-cueing computation (the reflectance stage of the rendering pipeline affects only surface primitives) and is mapped to a device color. The entire annotation text primitive is clipped if the *origin* is clipped. If the *origin* is not clipped, the annotation text is clipped according to text clipping rules. The connection line, if any, is clipped according to polyline clipping rules, except that modeling clipping is ignored. The current set of polyline attributes is used to render the connection line.

DATA STRUCTURES See **PEXlib.h.**

ERRORS

BadPEXRenderer
The specified renderer resource identifier is invalid.

BadPEXStructure
The specified structure resource identifier is invalid.

SEE ALSO

PEXSetTextFontIndex(3)
PEXSetTextPrecision(3)
PEXSetCharExpansion(3)
PEXSetCharSpacing(3)
PEXSetTextColorIndex(3)
PEXSetTextColor(3)
PEXSetATextHeight(3)
PEXSetATextUpVector(3)
PEXSetATextPath(3)
PEXSetATextAlignment(3)
PEXSetATextStyle(3)
PEXSetTextBundleIndex(3)