NAME

XGL_RAS_DEPTH – specifies the number of bits required to store the color of one pixel in a Raster

SYNOPSIS

#include <xgl/xgl.h>

Xgl_ras ras;

For XGL Memory Rasters only:
xgl_object_set(ras, XGL_RAS_DEPTH, Xgl_usgn32, 0);

For XGL Window and Memory Rasters:
xgl_object_get(ras, XGL_RAS_DEPTH, Xgl_usgn32 *);

DESCRIPTION

This attribute defines the number of bits that are used to specify the color of one pixel in an XGL Raster.

For XGL Window Rasters, **XGL_RAS_DEPTH**(3) is read-only.

For Memory Rasters, XGL_RAS_DEPTH is read-write and can be changed at any time. However, only four depths are currently supported — 1, 4, 8, and 32. Of these four, the 1-bit depth can be used only for creating stipple patterns. The 1-bit Memory Rasters as stipple fills are used to fill surface primitives (see **XGL_3D_CTX_SURF_BACK_FILL_STYLE**(3) and **XGL_CTX_SURF_FRONT_FILL_STYLE**(3)) or raster primitives (see **XGL_CTX_RASTER_FPAT**(3)).

The 1-bit Memory Rasters can be used as the source Raster in the **xgl_context_copy_raster**(3) operator, where the Destination Context can have either a 1-bit, 4-bit, 8-bit, or 32-bit Memory Raster Device.

The 4-bit, 8-bit and 32-bit depths can be used to create Memory Rasters that the XGL application will render into.

Note that changing the depth of a Memory Raster results in the loss of all pixel data within the Raster.

Note also that 32-bit Memory Rasters do not use the Context ROP mode or plane mask when rendering. If a 32-bit Memory Raster is dithered into an 8-bit Memory Raster or an 8-bit frame buffer, the ROP mode and plane mask are not used.

For Memory Rasters, the default value is 8. For Window Rasters, the default value is device-dependent.

SEE ALSO

xgl_object_set(3)
xgl_object_get(3)
xgl_context_copy_raster(3)
xgl_object_create(3)
XGL_3D_CTX_SURF_BACK_FILL_STYLE(3)
XGL_CTX_RASTER_FPAT(3)
XGL_CTX_SURF_FRONT_FILL_STYLE(3)

NOTES

32-bit deep Memory Rasters use just 24 bits to store color information. The remaining 8 bits are reserved for future use.