NAME

XGL_MEM_RAS – a Memory Raster Device object

OVERVIEW

XGL_MEM_RAS is the type of an object representing a Memory Raster Device. It is used as the *type* parameter value for **xgl_object_create**(3).

XGL graphics are rendered *on* a Device object. A specific Device object is the Memory Raster Device that draws into *pixels*, which are stored in the application's memory space. Because these pixels are not on a frame buffer, they are not visible to the operator. In contrast, a Window Raster Device object (see **xgl_object_create**(3) and XGL_WIN_RAS), which represents a hardware display device with a frame buffer, can display visible images.

After creating the Memory Raster Device, using **xgl_object_create**(3), the application can inquire about the values of its attributes by using the **xgl_object_get**(3) operator.

A Memory Raster Device object shares many attributes with the Window Raster Device object. These attributes have the common prefix XGL_RAS_. Attributes specific only to the Memory Raster Device have the prefix XGL_MEM_RAS_.

A brief overview of all of the attributes follows. For more information, see the man page for each attribute.

XGL_DEV_COLOR_MAP(3)
: Specifies the Color Map Object used by the Memory Raster Device object.

XGL_DEV_COLOR_TYPE(3)
: Specifies the color type (that is, the color model) of the Raster. It can be set only once, when the Memory Raster Device object is created. The two alternatives are: XGL_COLOR_INDEX and XGL_COLOR_RGB.

XGL_DEV_CONTEXTS(3)
XGL_DEV_CONTEXTS_NUM(3)
: These attributes return the Context object handles used by the Memory Raster Device object.

XGL_DEV_MAXIMUM_COORDINATES(3)
: Returns the maximum coordinate values for the Memory Raster Device object.

XGL_DEV_PICK_BUFFER_SIZE(3)
: Returns the maximum pick buffer size that a Memory Raster Device object can support efficiently.

XGL_DEV_REAL_COLOR_TYPE(3)
: Specifies the real color space used by the Memory Raster Device object.

XGL_MEM_RAS_IMAGE_BUFFER_ADDR(3)
: Returns the starting address of the block of memory set used by the Memory Raster Device.

XGL_MEM_RAS_Z_BUFFER_ADDR(3)
: Returns the starting address of the block of memory set used by the Z buffer of a memory raster.

XGL_OBJ_APPLICATION_DATA(3)
This attribute allows the application program to store its own data with an object.

XGL_OBJ_SYS_STATE(3)
This read-only attribute returns the System State that was used to create the object.

XGL_OBJ_TYPE(3)
This read-only attribute defines the type of an XGL object.

XGL_RAS_DEPTH(3)
XGL_RAS_HEIGHT(3)
XGL_RAS_WIDTH(3)

These attributes give the size of the Memory Raster Device. They can be set at any time, however. If any of them are changed after the Raster has been created, all pixel data currently stored in the Raster is lost. The depth of a Memory Raster Device can be 1, 4, 8, or 32, which is the number of *bit-planes* in the Raster (that is, the number of bits that are used to specify the color of one pixel).

XGL_RAS_RECT_LIST(3)
XGL_RAS_RECT_NUM(3)

These attributes define the number and list of clip rectangles in the application-specified clip list.

XGL_RAS_SOURCE_BUFFER(3)
This attribute specifies the buffer to be used as source during raster operations.

XGL does not render into 1-bit-deep Memory Rasters. This means that you cannot use a 1-bit Memory Raster as the device for a Context. The 1-bit Memory Rasters are used for stipple-filled polygons and for the xgl_context_copy_raster function, where the source Raster is a 1-bit Memory Raster, and the device for the destination context is either a 4-bit, 8-bit or 32-bit Raster.

NOTES

32-bit Memory Rasters with an XGL_DEV_COLOR_TYPE of XGL_COLOR_INDEX are not supported. When creating a 32-bit Memory Raster, the application should set XGL_DEV_COLOR_TYPE to XGL_COLOR_RGB.

4-bit and 8-bit Indexed (that is, XGL_DEV_COLOR_TYPE is XGL_COLOR_INDEX) Memory Rasters store their pixel values as application-defined color values (that is, the indices are exactly those specified, either explicitly or implicitly, by the application).