NAME

XGL_TMAP_PARAM_INFO – specifies additional information used in texture parameterization

SYNOPSIS

```
#include <xgl/xgl.h>
Xgl_tmap            tmap;
xgl_object_set(tmap, XGL_TMAP_PARAM_INFO, Xgl_texture_param_info, 0);
xgl_object_get(tmap, XGL_TMAP_PARAM_INFO, Xgl_texture_param_info *);
```

DESCRIPTION

This attribute is used when the attribute value of **XGL_TMAP_PARAM_TYPE**(3) is other than XGL_TEXTURE_PARAM_EXPLICIT.

The field *project* of type Xgl_texture_projection in the *Xgl_texture_param_info* structure is used in generating the texture coordinates.

If **XGL_TMAP_PARAM_TYPE**(3) is XGL_TEXTURE_PARAM_REFLECT_SPHERE_EC or XGL_TEXTURE_-PARAM_REFLECT_SPHERE_WC, then the reflection matrix as specified by the field *geom_orient_matrix* of type Xgl_matrix_f3d in the *Xgl_texture_projection* structure is used in the texture generation.

XGL_TEXTURE_PARAM_REFLECT_SPHERE_EC

The reflection vector $\{ s_0, s_1, s_2 \}$ is calculated as follows:

s = i – 2n(n dot i)

where *s* is the reflection vector, *i* is the unit vector, and *n* is the vertex normal from the eyepoint to the vertex. This calculation is performed after the vertex and its normal have been transformed to the Eye Coordinate space. The reflection vector is then transformed by the reflection matrix. After this, the texture coordinates are derived as follows:

$$t_0 = \frac{s_0}{m} + \frac{1}{2}$$

$$t_1 = \frac{s_1}{m} + \frac{1}{2}$$

where: $m = 2 \times \sqrt{s_0^2 + s_1^2 + (s_2 + 1)^2}$

XGL_TEXTURE_PARAM_REFLECT_SPHERE_WC

The reflection vector is calculated as:

s = –(i – 2n(n ∗ i))

where *s* is the reflection vector, *n* is the vertex normal and *i* is the unit vector from the eyepoint to the vertex. The calculation is performed in WC space, after the vertex and its normal have been transformed to WC space. After the reflection vector is calculated, the vector is transformed by the reflection matrix. After this, the texture coordinates are derived as follows:

$$t_0 = 0.75 : s_0 = 0.0, s_2 > 0.0$$

$$= 0.25 : s_0 = 0.0, s_2 \le 0.0$$

$$= \frac{-atan\left[\frac{s_2}{s_0}\right]}{2\pi} + 0.5 : s_0 < 0.0$$

$$= \frac{-atan\left[\frac{s_2}{s_0}\right]}{2\pi} + 1.0 : s_0 > 0.0, s_2 > 0.0$$

$$= \frac{-atan\left[\frac{s_2}{s_0}\right]}{2\pi} + 0.0 : s_0 > 0.0, s_2 \le 0.0$$

$$t_1 = \frac{asin\,(s_1)}{\pi + 0.5}$$

where: $\left[\frac{-\pi}{2} \le atan\ (r) \le \frac{\pi}{2}\right]$ and $\left[\frac{-\pi}{2} \le asin\ (r) \le \frac{\pi}{2}\right]$

If **XGL_TMAP_PARAM_TYPE**(3) is XGL_TEXTURE_PARAM_LINEAR_EC, then the projection values as specified by the field *linear* of *Xgl_texture_projection* is used.

XGL_TEXTURE_PARAM_LINEAR_EC

The vertex { V_x, V_y, V_z } is transformed into the Eye Coordinate space and the texture coordinates are derived as follows:

$$t_0 = V_x * P_0' * x + V_y * P_0' * y + V_z * P_0' * z + P_0' * w$$

$$t_0 = V_x * P_1' * x + V_y * P_1' * y + V_z * P_1' * z + P_1' * w$$

where P_0' and P_1' are derived by transforming the values specified in *Xgl_texture_param_info* by the Eye Coordinate to Model Coordinate matrix.

The default value is NULL.

SEE ALSO

xgl_object_set(3)
xgl_object_get(3)
XGL_TMAP_PARAM_TYPE(3)