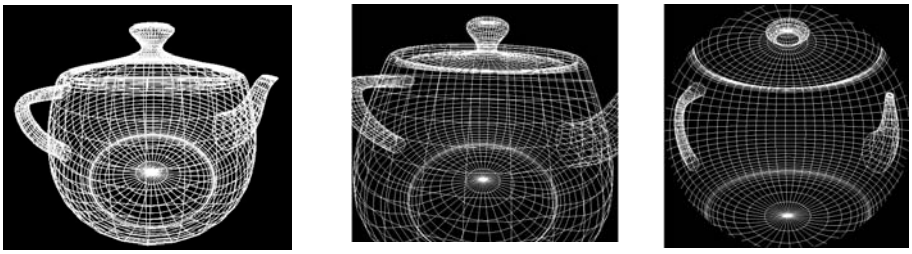
GPU Programming for Dome Projection

Mike Bailey
mjb@cs.oregonstate.edu

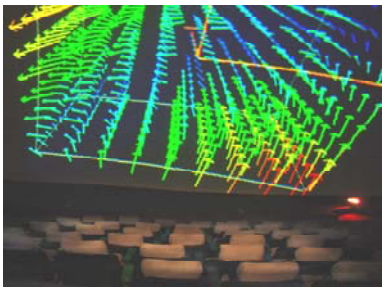
Purpose: Dome projection requires a fisheye projector lens, which necessarily distorts the image. The trick is to pre-distort the image in the other direction during the rendering process so that the projected image looks correct. This could be done in software or in multi-pass hardware, but this is slow. Instead, we are doing it in a single-pass GPU program.

```
const float PI =  3.14159265;
const float C1 = ??;
const float C2 = ??;
const float C3 = ??;
const float C4 = ??;
const float C5 = ??;

void
main( void )
{
        float phi;
        vec4 pos = gl_ModelViewMatrix * gl_Vertex;
        float rxy = length( pos.xy );

        if( rxy != 0.0 )
        {
                float phi = atan( rxy, -pos.z );
                // float lens_radius = phi / (PI/2.);
                float lens_radius = phi* (C1 + phi* (C2 + phi* (C3 + phi* (C4 + phi*C5) ) ) );
                        pos.xy *= ( lens_radius / rxy );
        }

        gl_Position = gl_ProjectionMatrix * pos;
}
```

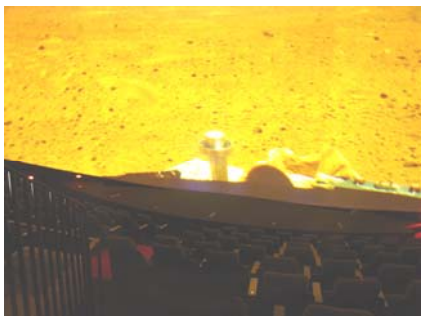
GLSL Shader Code



Teapots on a Monitor: Undistorted, Dome-distorted, Dome-distorted and Zoomed-in



Fluid flow visualization in a dome



Mars panoramic display in a dome

November 23, 2005