

CMSC 435/634  
Introductory Computer Graphics  
Renderman  
Penny Rheingans  
UMBC

## Renderman Environment

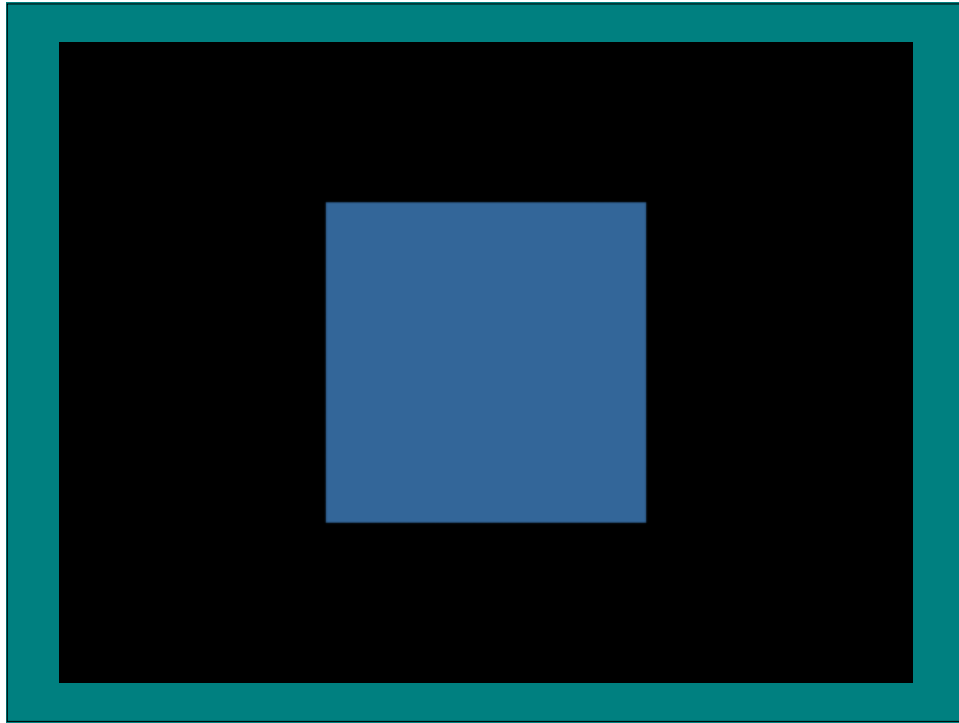
- Development Cycle
  - write C code
  - make → executable
  - run program → TIFF file
  - display imagefile
- Installed on gl
  - rlogin, DISPLAY, etc
  - source .pixie\_csh

## Minimal Program (simple square)

```
#include "ri.h"
RtPoint Square[4]={{.5,.5,.5},{.5,-.5,.5},{-.5,-.5,.5},{-.5,.5,.5}};
static RtColor Color = {.2, .4, .6};
static RtColor Color2 = {.0, .0, .6};
main () {
    RiBegin(RI_NULL); /* start the renderer */
    RiColor (Color2); /* declare color */
    RiWorldBegin();
        RiSurface ("constant", RI_NULL);
        RiColor (Color); /* declare color */
        RiPolygon (4, /* declare the square */
            RI_P, (RtPointer) Square, RI_NULL);
    RiWorldEnd();
    RiEnd();
}
```

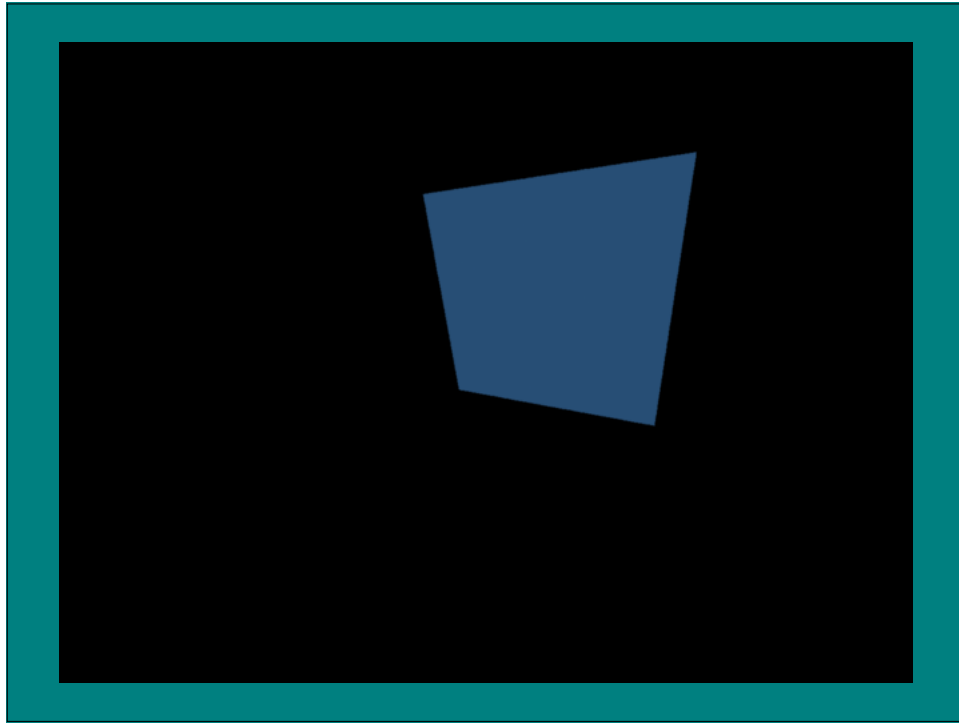
## Square RIB

```
##RenderMan RIB-Structure 1.0
version 3.03
Color [0 0 0.6]
WorldBegin
Surface "constant"
Color [0.2 0.4 0.6]
Polygon "P" [0.5 0.5 0.5 0.5 -0.5 0.5 -0.5 -0.5
    0.5 -0.5 0.5 0.5]
WorldEnd
```



## Refined Program (refined square)

```
main () {
  RiBegin(RI_NULL); /* start the renderer */
  RiLightSource ("distantlight", RI_NULL);
  RiProjection("perspective", RI_NULL);
  RiTranslate(0.0,0.0,1.0);
  RiRotate (40.0, -1.0, 1.0, 0.0);
  RiWorldBegin();
    RiSurface ("matte", RI_NULL);
    RiColor (Color); /* declare color */
    RiPolygon (4, /* declare the square */
      RI_P, (RtPointer) Square, RI_NULL);
  RiWorldEnd();
  RiEnd();
}
```

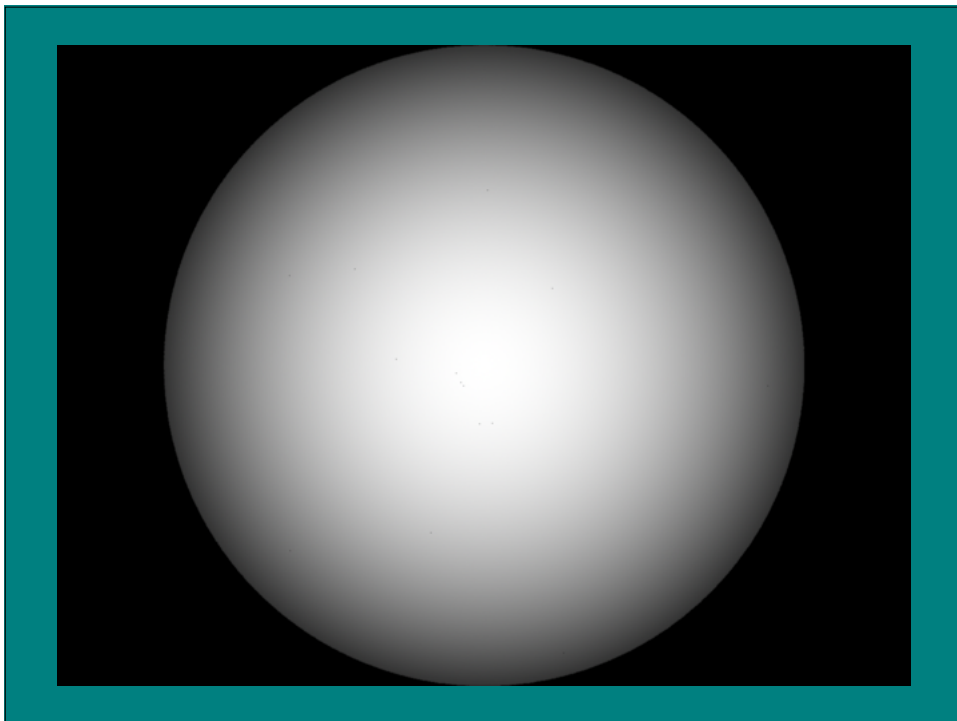


## Alternative Primitive (sphere)

```
main () {
  RtFloat radius=1.0, zmin=-1.0, zmax=1.0, thetamax=360;
  RiBegin(RI_NULL); /* start the renderer */
  RiLightSource ("distantlight", RI_NULL);
  RiProjection("perspective", RI_NULL);
  RiTranslate(0.0,0.0,1.0);
  RiRotate (40.0, -1.0, 1.0, 0.0);
  RiWorldBegin();
  RiSurface ("matte", RI_NULL);
  RiColor (Color); /* declare color */
  RiSphere (radius, zmin, zmax, thetamax, RI_NULL);
  RiWorldEnd();
  RiEnd();
}
```

## Sphere RIB

```
WorldBegin
  Translate 0 0 2
  Sphere 1 -1 1 360
WorldEnd
```

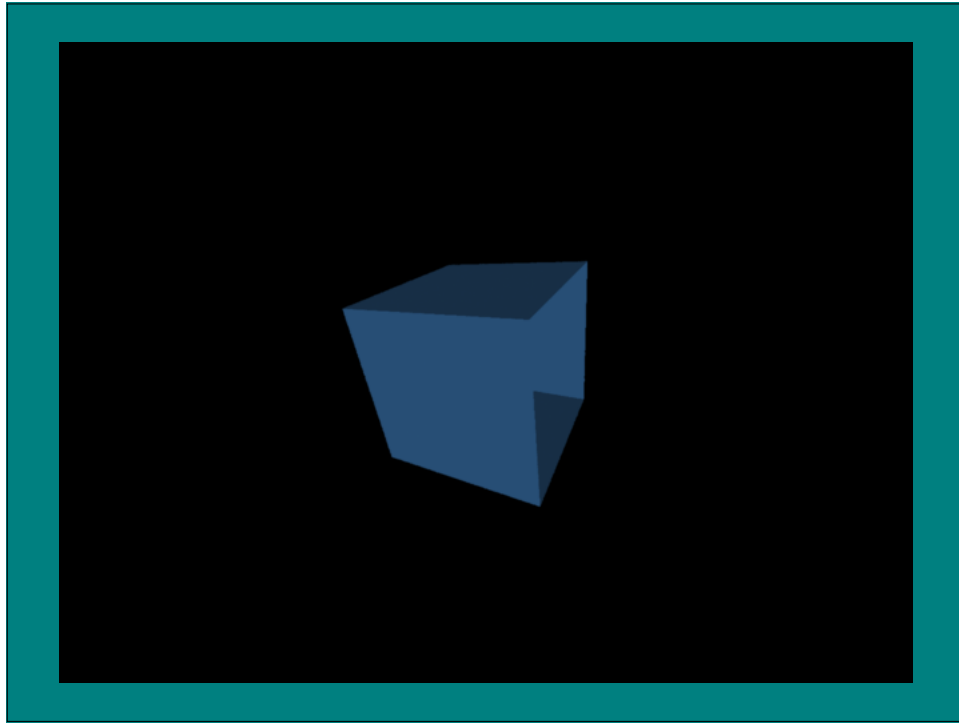


## Basic cube program

```
main () {
    RiBegin(RI_NULL); /* start the renderer */
    RiLightSource ("distantlight", RI_NULL);
    RiProjection("perspective", RI_NULL);
    RiTranslate(0.0,0.0,1.0);
    RiRotate (40.0, -1.0, 1.0, 0.0);
    RiWorldBegin();
        RiSurface ("matte", RI_NULL);
        RiColor (Color); /* declare color */
        UnitCube();
    RiWorldEnd();
    RiEnd();
}
```

## Cube Procedure

```
#define L -.5           #define R .5
#define D -.5         #define U .5
#define N -.5         #define F .5
UnitCube () {
    static RtPoint Cube[6][4] = {
        {{L,D,F}, {R,D,F}, {R,D,N}, {L,D,N}}, // bottom
        {{L,D,F}, {L,U,F}, {L,U,N}, {L,D,N}}, // left
        {{R,U,N}, {L,U,N}, {L,U,F}, {R,U,F}}, // top
        {{R,U,N}, {R,U,F}, {L,U,F}, {L,D,F}}, // right
        {{R,D,F}, {R,U,F}, {L,U,F}, {L,D,F}}, // far
        {{L,U,N}, {R,U,N}, {R,D,N}, {L,D,N}}; //near
    int i;
    for (i = 0; i < 6; i++)
        RiPolygon(4,RI_P, (RtPointer)Cube[i],RI_NULL);
}
```



## Refined Cube Procedure

```
UnitCube () {  
    RiTransformBegin();  
    RiPolygon(4,RI_P,(RtPointer)square, RI_NULL);  
    RiRotate(90.0, 0.0, 1.0, 0.0); // right face  
    RiPolygon(4,RI_P,(RtPointer)square, RI_NULL);  
    RiRotate(90.0, 0.0, 1.0, 0.0); // near face  
    RiPolygon(4,RI_P,(RtPointer)square, RI_NULL);  
    RiRotate(90.0, 0.0, 1.0, 0.0); // left face  
    RiPolygon(4,RI_P,(RtPointer)square, RI_NULL);  
    RiTransformEnd();  
    RiTransformBegin(); // bottom face  
    RiRotate(90.0, 1.0, 0.0, 0.0);  
    RiPolygon(4,RI_P,(RtPointer)square, RI_NULL);  
    RiTransformEnd();  
    RiTransformBegin(); // top face  
    RiRotate(-90.0, 1.0, 0.0, 0.0);  
    RiPolygon(4,RI_P,(RtPointer)square, RI_NULL);  
    RiTransformEnd();  
}
```

## Some other features

- Lights
- Quadric surfaces: disks, cones, cylinders
- Textures from images
- Shaders

## Assignment 1

- Use renderman to model and render a plate of veggies