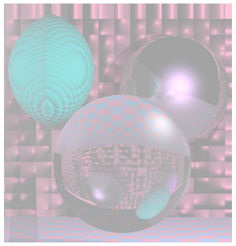
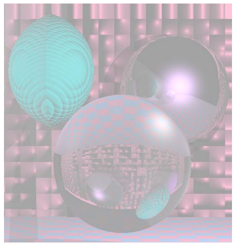
CS4300
Computer Graphics

Prof. Harriet Fell
Fall 2012
Lecture 22 – October 25 ,2012



Today's Topics

- Poly Mesh
  - Hidden Surface Removal
  - Visible Surface Determination
  - More about the First 3D Project
  - First Lighting model

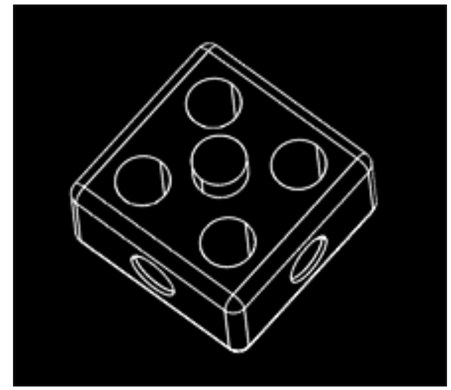


Rendering a Polymesh

- Scene is composed of triangles or other polygons.
- We want to view the scene from different view-points.
  - Hidden Surface Removal
    - Cull out surfaces or parts of surfaces that are not visible.
  - Visible Surface Determination
    - Head right for the surfaces that are visible.
    - Ray-Tracing is one way to do this.

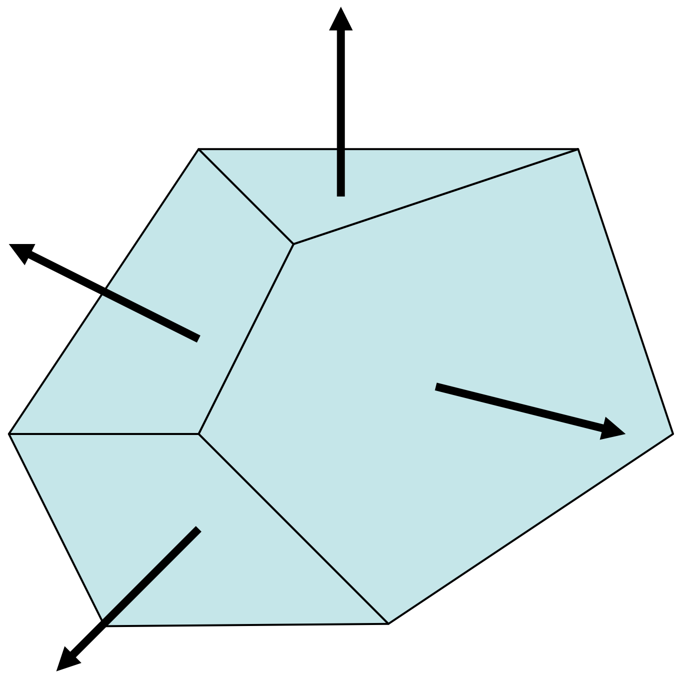
Wireframe Rendering

Hidden-
Line
Removal



Hidden-
Face
Removal

Convex Polyhedra



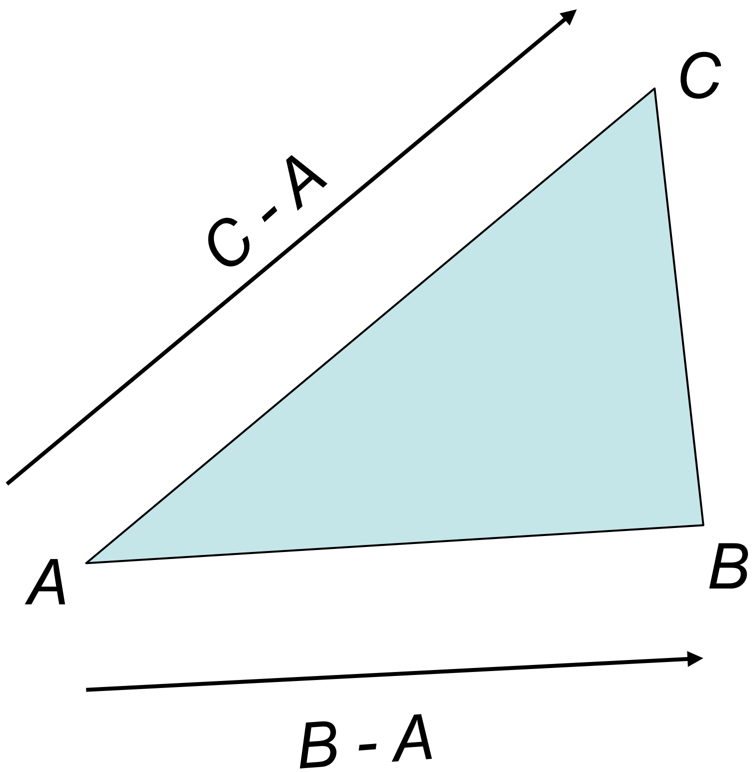
We can see a face if and only if its normal has a component toward us.

$$N \cdot V > 0$$

V points from the face toward the viewer.

N point toward the outside of the polyhedra.

Finding N



$N = (B - A) \times (C - A)$

is a normal to the triangle that points toward you.

$$\frac{N}{\|N\|}$$

is a unit normal that points toward you.

Code for *N*

```
private Vector3d findNormal(){
        Vector3d u = new Vector3d();
        u.scaleAdd(-1, verts[0], verts[1]);
        Vector3d v = new Vector3d();
        v.scaleAdd(-1, verts[0], verts[2]);
        Vector3d uxv = new Vector3d();
        uxv.cross(u, v);
        return uxv;
}
```

Finding V

- Since we are just doing a simple orthographic projection, we can use

  $V = k = (0, 0, 1).$
- Then

  $N \bullet V$ = the z component of N

```
public boolean faceForward() {
  return (normal.z > 0);
}
```

Find *L*

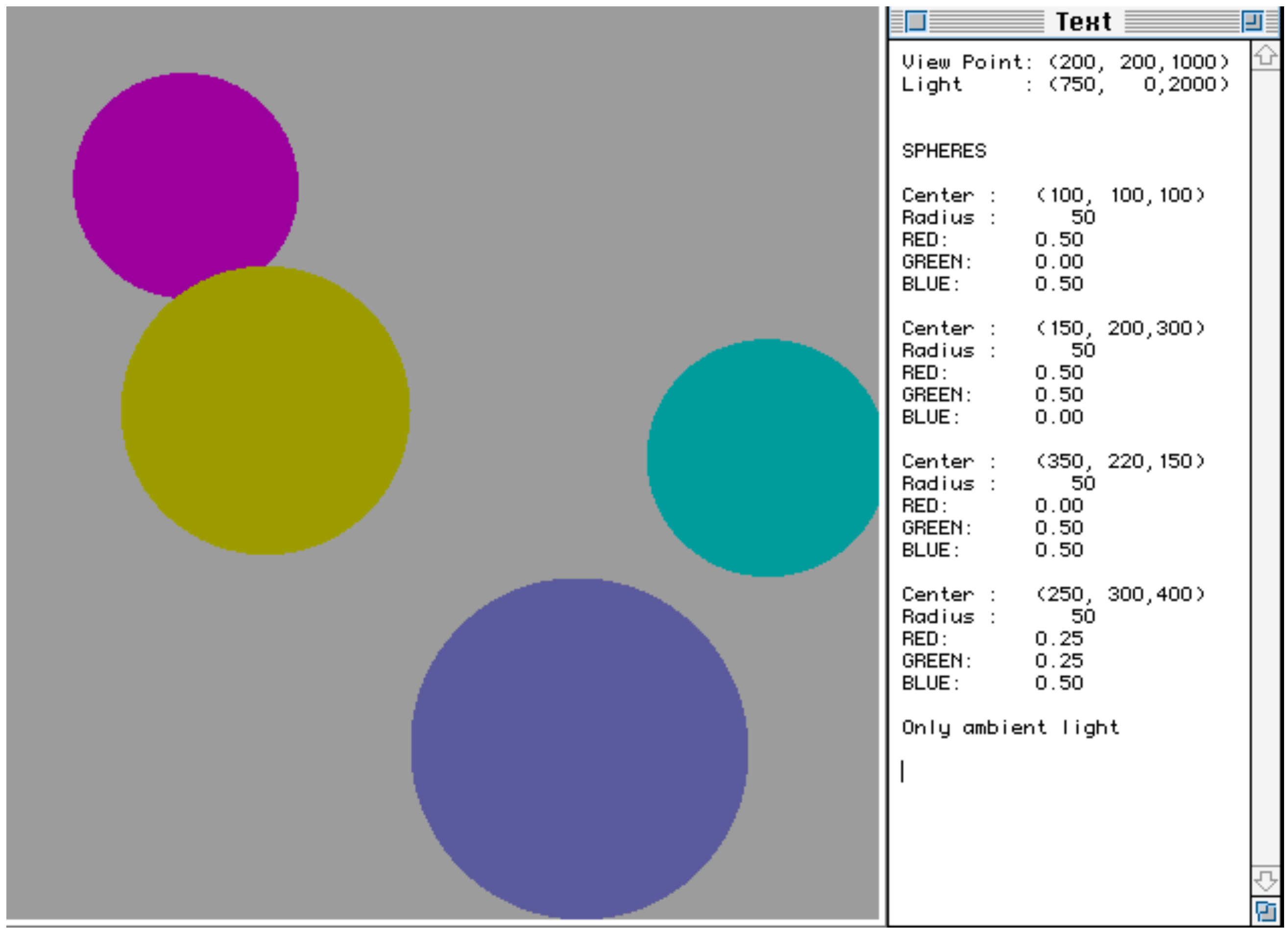
- *L* is a unit vector from the point you are about to render toward the light.
- For the faceted icosahedron use the center point of each face.
  - $cpt = (A + B + C)/3$

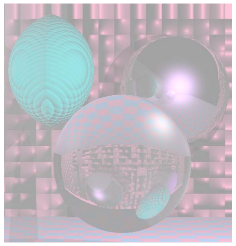
First Lighting Model

- Ambient light is a global constant *ka*.
  - Try *ka* = .2.
  - If a visible object S has color (S_R, S_G, S_B) then the ambient light contributes

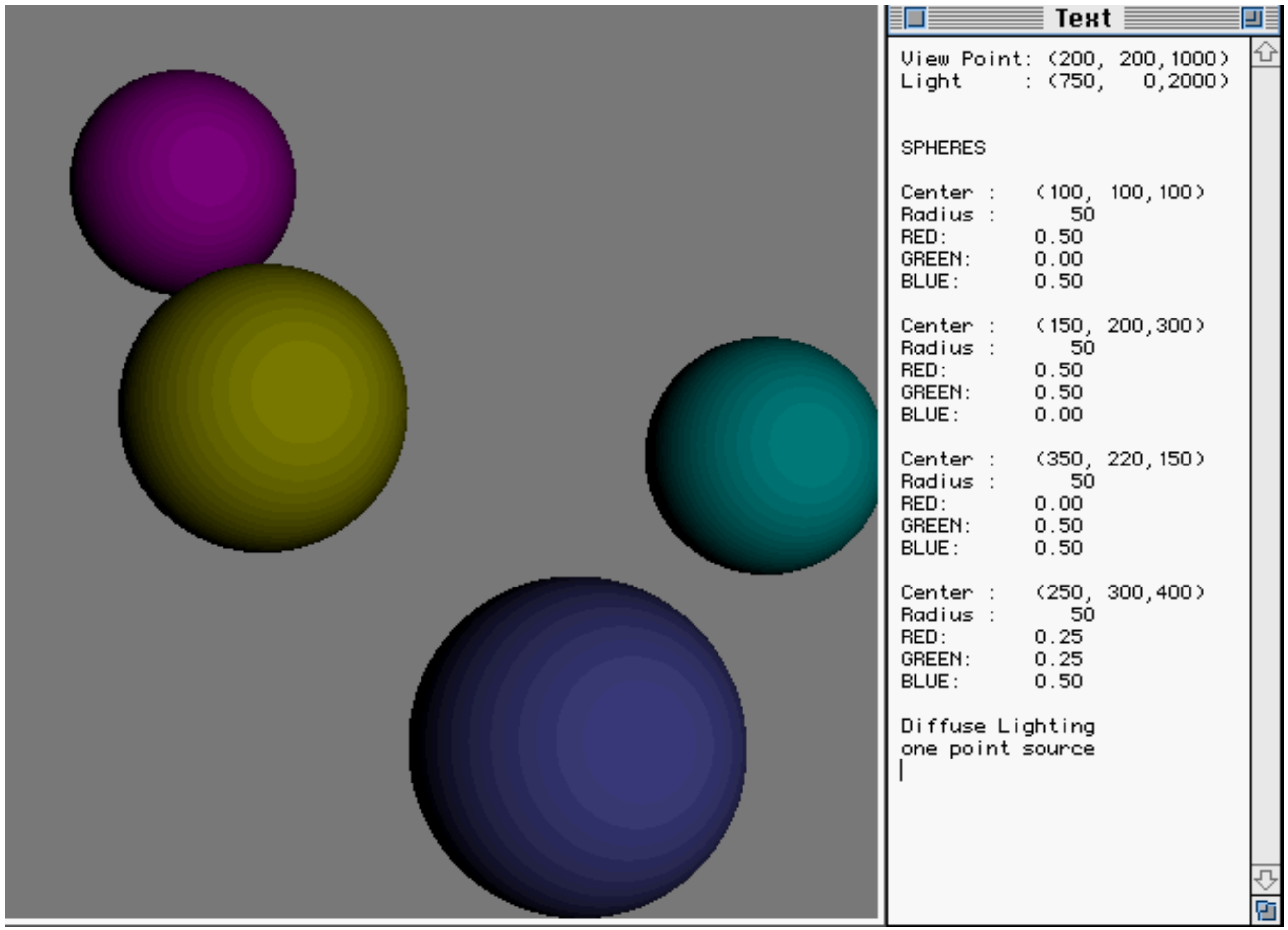
    $(.2^* S_R, .2^* S_G, .2^* S_B)$.
- Diffuse light depends of the angle at which the light hits the surface. We add this to the ambient light.
- We will also add a spectral highlight.

Visible Surfaces
Ambient Light





Diffuse Light

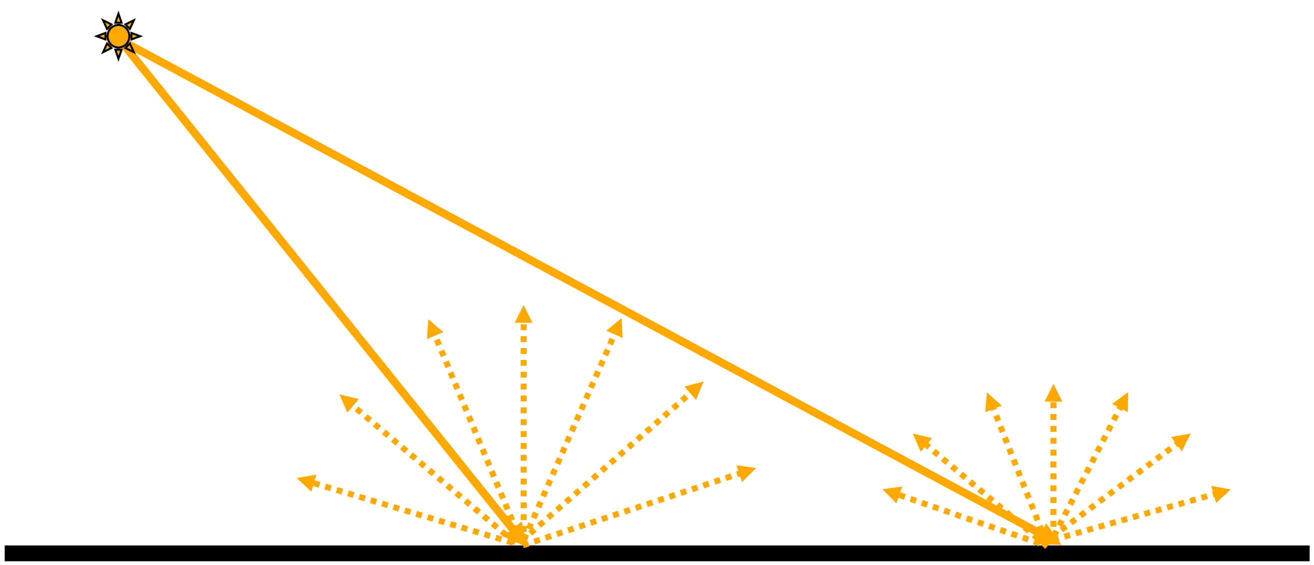


Lambertian Reflection Model
Diffuse Shading

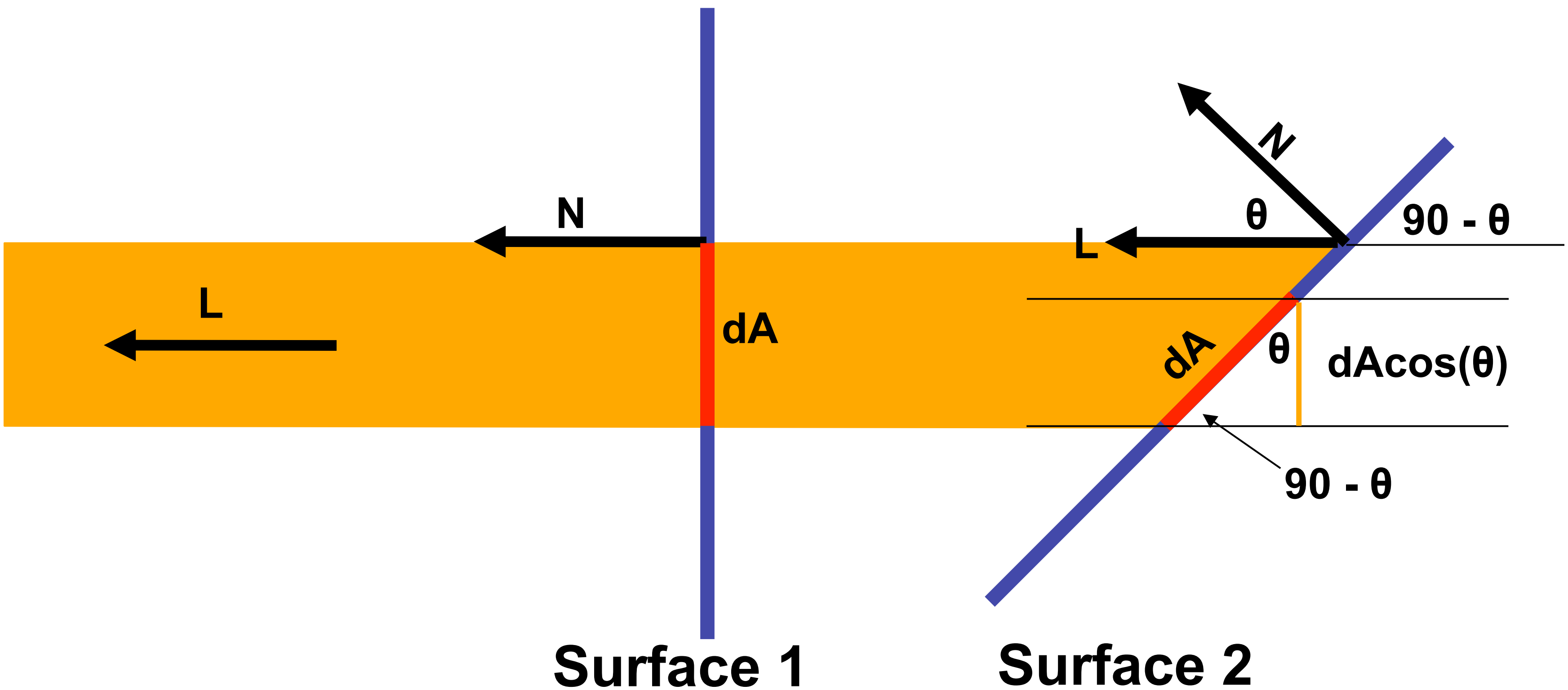
- For matte (non-shiny) objects
- Examples
  - Matte paper, newsprint
  - Unpolished wood
  - Unpolished stones
- Color at a point on a matte object does not change with viewpoint.

Physics of Lambertian Reflection

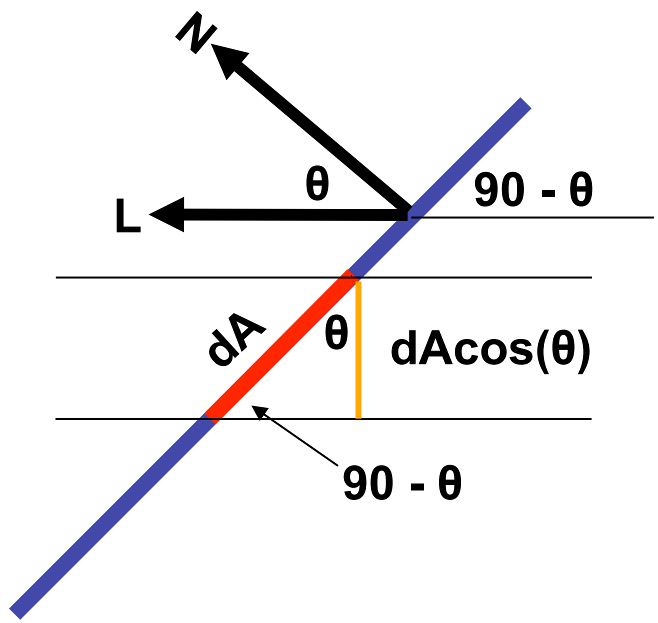
- Incoming light is partially absorbed and partially transmitted equally in all directions



Geometry of Lambert's Law



cos(θ)=N•L

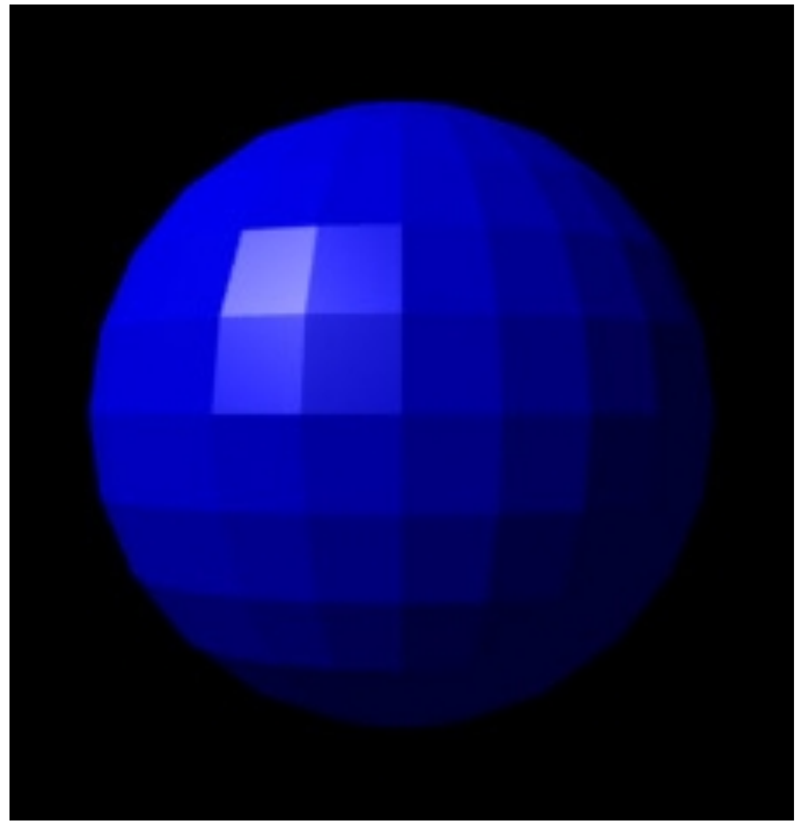


Surface 2

Cp= ka (SR, SG, SB) + kd **N•L** (SR, SG, SB)

Flat Shading

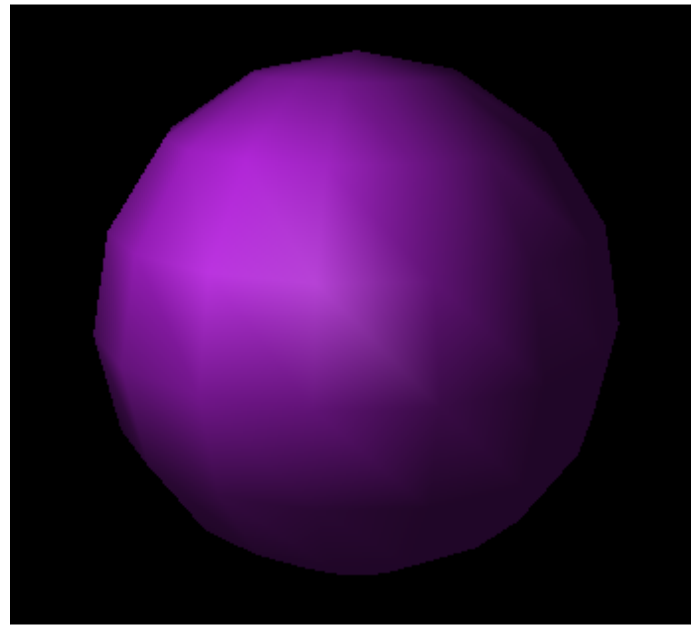
- A single normal vector is used for each polygon.
- The object appears to have facets.

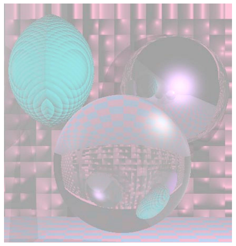


http://en.wikipedia.org/wiki/Phong_shading

Gouraud Shading

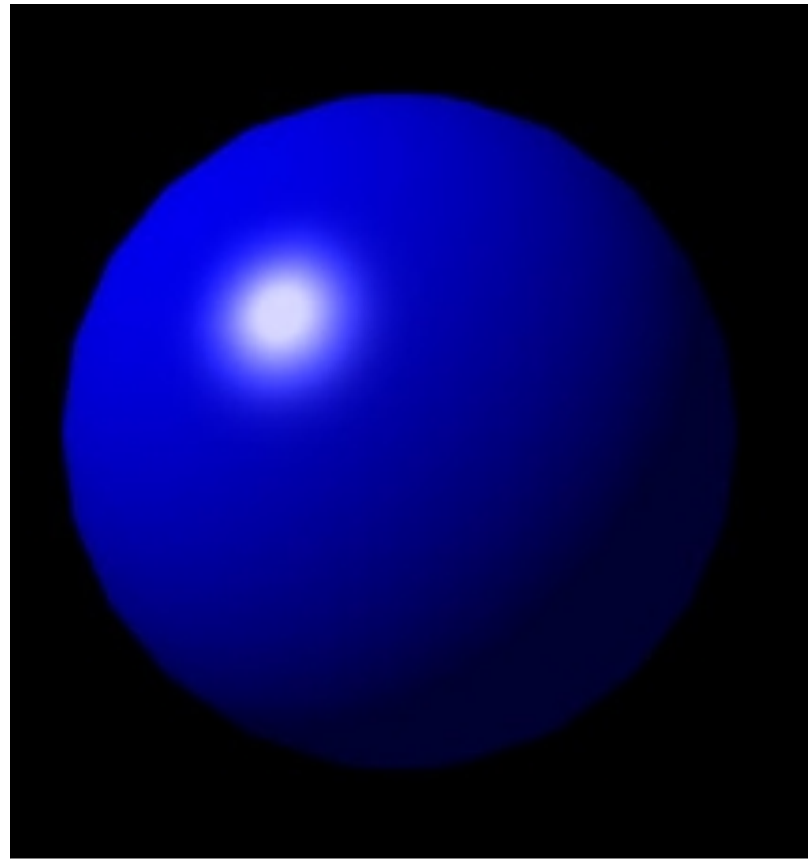
- Average the normals for all the polygons that meet a vertex to calculate its surface normal.
- Compute the color intensities at vertices base on the Lambertian diffuse lighting model.
- Average the color intensities across the faces.





Phong Shading

- Gouraud shading lacks specular highlights except near the vertices.
- Phong shading eliminates these problems.
- Compute vertex normals as in Gouraud shading.
- Interpolate vertex normals to compute normals at each point to be rendered.
- Use these normals to compute the Lambertian diffuse lighting.



http://en.wikipedia.org/wiki/Phong_shading