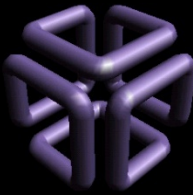




Computação Gráfica:

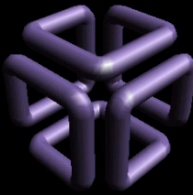
Aula 10:
Modelo de Phong
Tipos de Raytracing

Prof. Dr. rer.nat. Aldo von Wangenheim



Conteúdo desta Aula

- **Modelo de Iluminação de Phong**
 - Luz Difusa
 - Luz Especular
 - Luz Ambiente
 - Equação de Phong
- **Técnicas de Iluminação**
 - Raytracing
 - Raycasting
 - Rayshading
 - Pixel Shading



Modelando a Iluminação de um Objeto

- O modelo de iluminação de Phong [PHO75] é o modelo de iluminação provavelmente mais utilizado em computação gráfica hoje em dia.
- É uma simplificação da teoria de como a luz age no mundo real.

[PHO75] - Phong, Bui-Tuong, "Illumination for Computer-Generated Pictures", Communications of the ACM, vol. 18, no. 3, pp. 311-317, 1975.



Modelando a Iluminação de um Objeto

- No modelo de Phong, a luz em qualquer ponto é composta por três componentes: **luz difusa**, **luz especular** e **luz ambiente**.
 - Essas três componentes são aditivas e determinam o aspecto final da iluminação e da cor de um determinado ponto na cena ou da superfície de um determinado polígono plano contido nela.

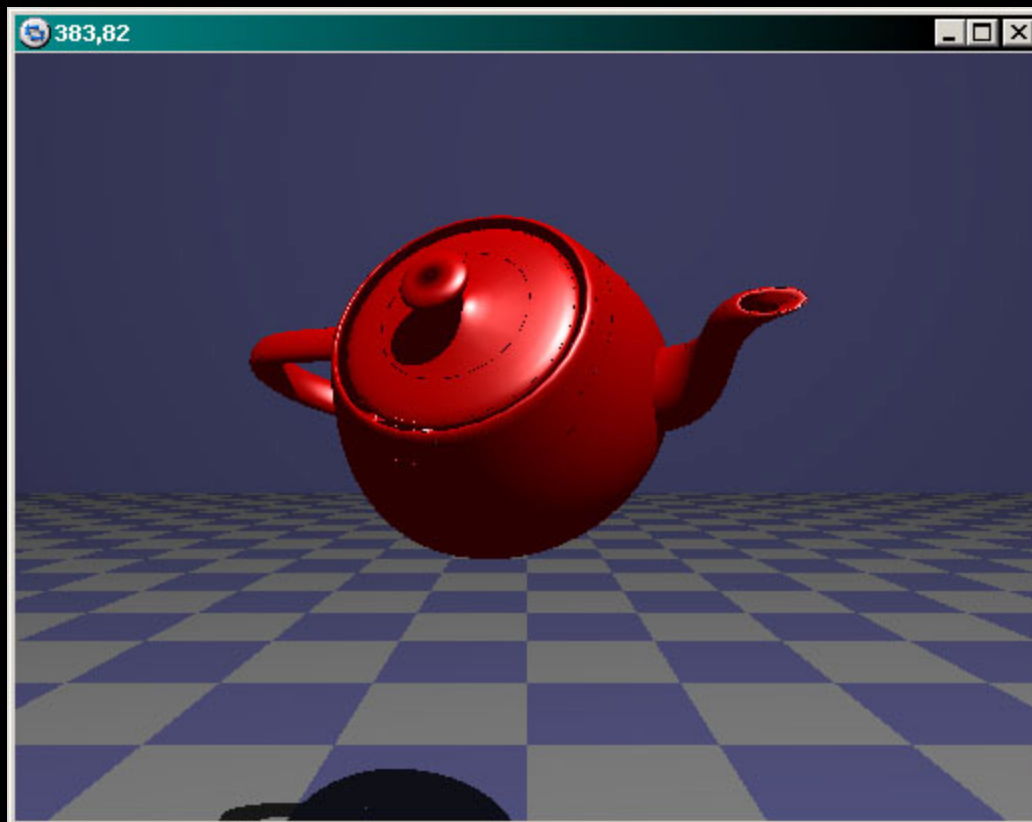


The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC





Modelando a Iluminação de um Objeto

- Os componentes essenciais no modelo de Phong são:
 - A posição dos pontos emissores de luz e suas propriedades
 - As características dos objetos do mundo, do ponto de vista de sua interação com as luzes
 - A posição do observador no espaço

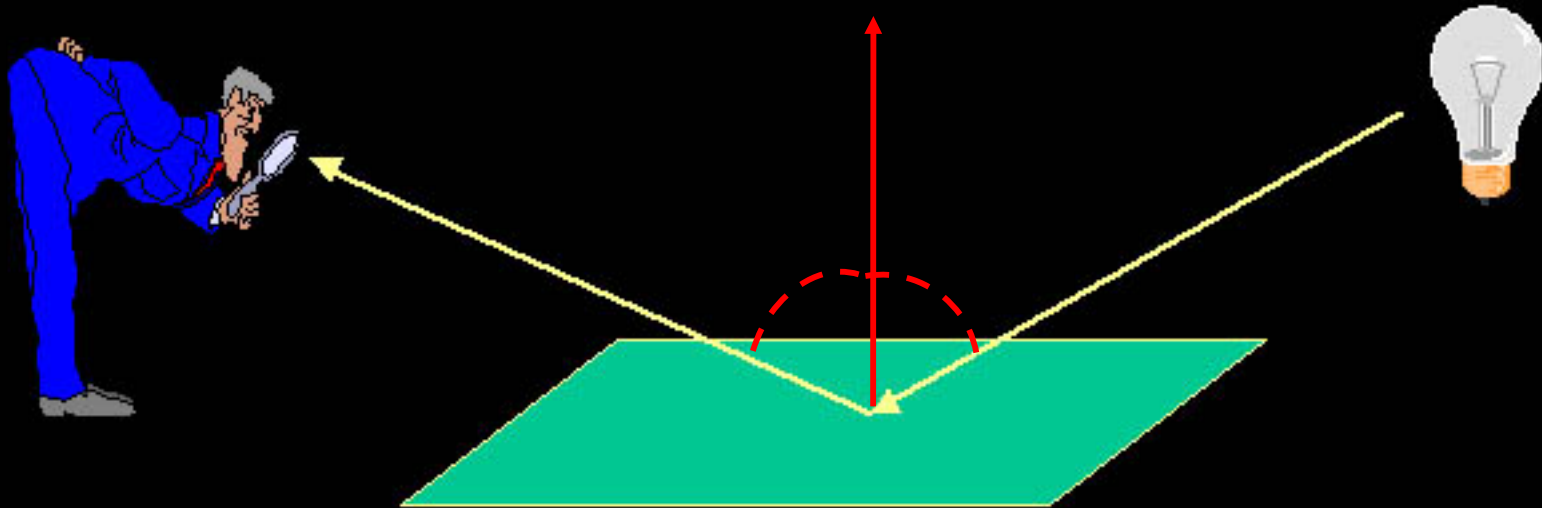


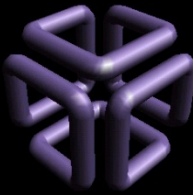
The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

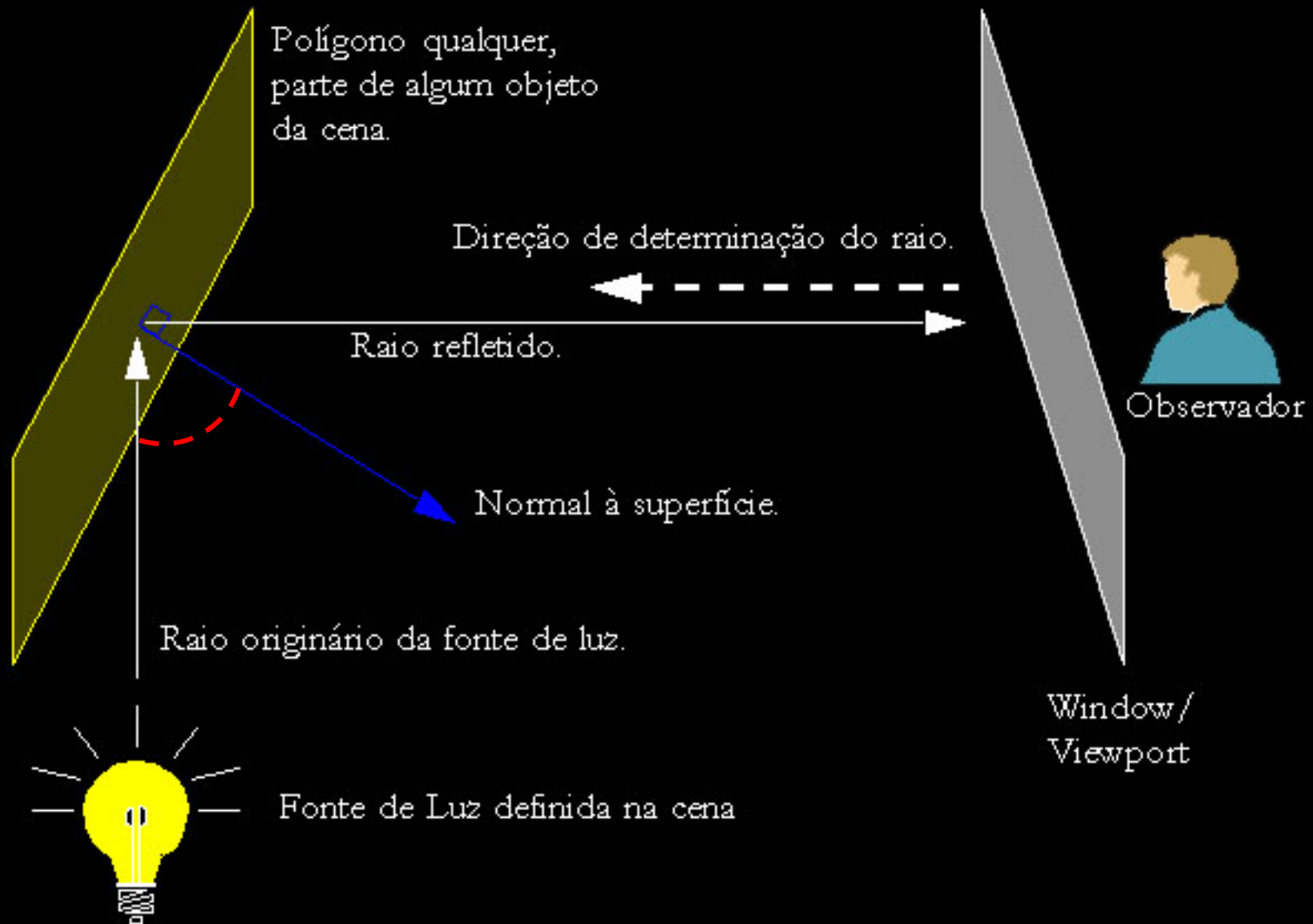
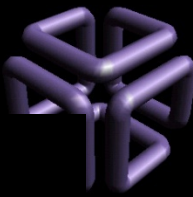
Curso de Ciência da Computação
INE/CTC/UFSC

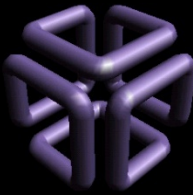




Luz Difusa

- A componente difusa do modelo de Phong representa reflexões que não são direcionais em sua natureza.
 - Cada objeto possui características de reflectância difusa, codificadas como um coeficiente de reflectância difusa, que determinam quanta luz é refletida por sua superfície.
 - A quantidade de luz difusa refletida é independente da direção da qual a superfície é vista, uma vez que superfícies que refletem de forma difusa refletem de forma igual em todas as direções.
 - A intensidade da reflexão difusa varia apenas com o cosseno do ângulo entre a normal da superfície e a fonte de luz, que nos indica a quantidade de luz que incide sobre aquele ponto.





Luz Difusa - Phong

$$I_{dif} = I_L r_{dif} (\vec{v}_L \cdot \vec{v}_n)$$

I_{dif} - Intensidade da Luz Difusa

I_L - Intensidade da Fonte de Luz

r_{dif} - Coeficiente de Reflectividade Difusa da Superfície

\vec{v}_L - Vetor Normalizado apontando para Fonte de Luz

\vec{v}_n - Normal da Superfície

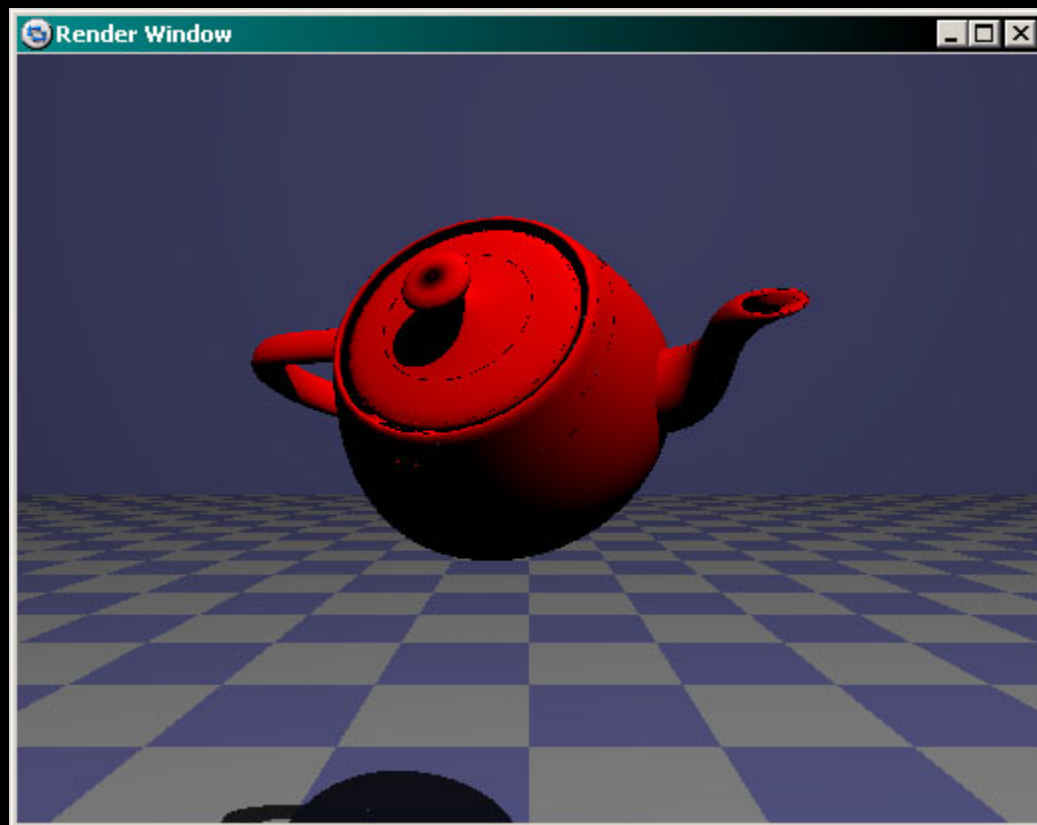


The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

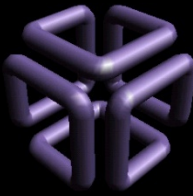
Curso de Ciência da Computação
INE/CTC/UFSC





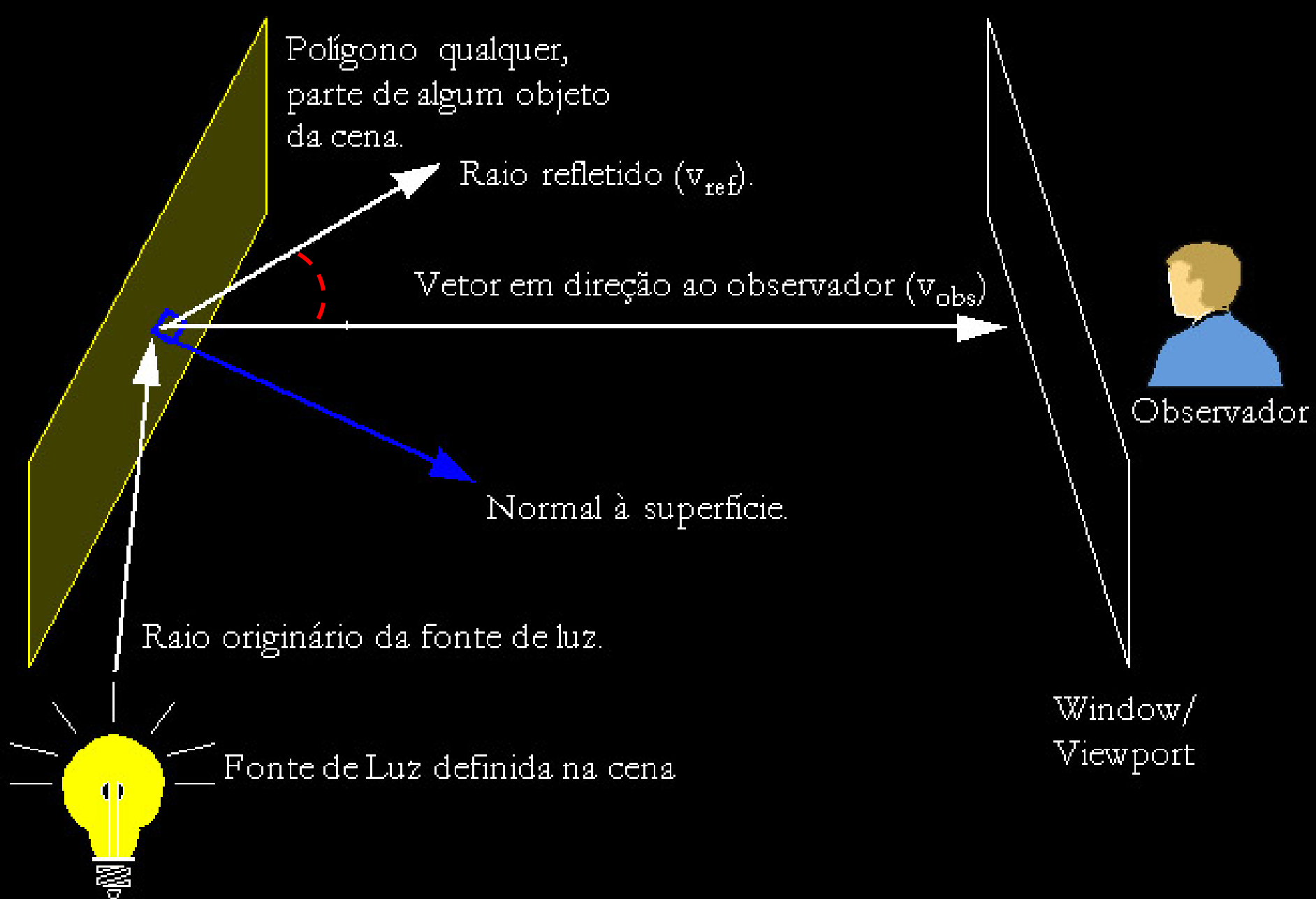
Luz Especular

- A componente especular de um objeto simula reflexos extremamente direcionais, sem que a cor dos raios seja afetada pela cor da superfície, como num espelho.
 - Indica o quão "brilhante" este objeto é.
 - Ela serve para simular objetos de superfície muito polida e que refletem a luz, como objetos metálicos ou esmaltados.
 - Baseia em **coeficiente de reflectividade especular** que determina quanta luz é refletida por este objeto.



Luz Especular

- A componente especular de um objeto simula reflexos extremamente direcionais, sem que a cor dos raios seja afetada pela cor da superfície, como num espelho.
 - A intensidade da reflexão especular é proporcional ao cosseno do ângulo entre a direção de visada e a direção de reflexão da luz.
 - **Expoente especular** determina o quão rápido o reflexo decai quando o ângulo de visada se afasta do ângulo de reflexão.





Luz Especular - Phong

$$I_{espec} = I_L r_{espec} (\vec{v}_{ref} \cdot \vec{v}_{obs})^s$$

I_{espec} - Intensidade da Luz Especular

I_L - Intensidade da Fonte de Luz

r_{espec} - Coef. de Reflectividade Especular da Superfície

\vec{v}_{ref} - Vetor normalizado da direção de reflexão

\vec{v}_{obs} - Vetor normalizado da direção do observador

s - Expoente especular

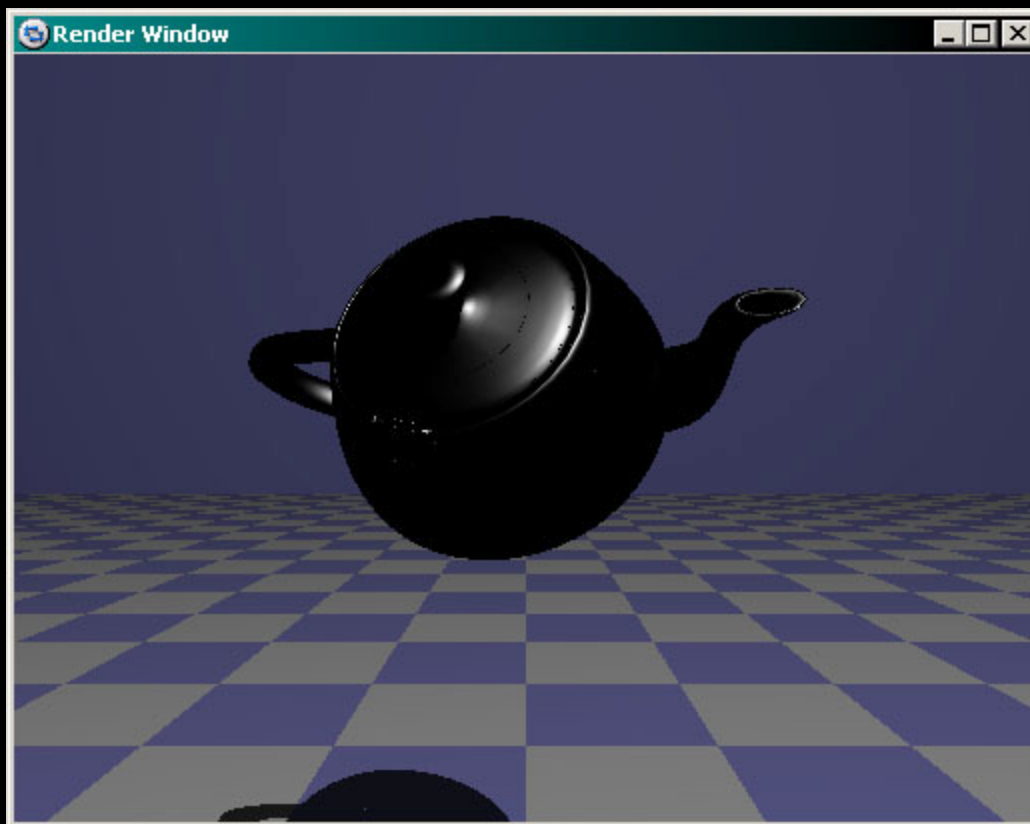
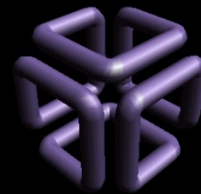


The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC





Luz Ambiente

- A componente de luz ambiente do modelo de Phong foi adicionada para levar em consideração luz gerada por interações inter-objetos.
 - Em ambientes reais, superfícies que não são iluminadas diretamente geralmente não se encontram completamente escuras.
 - Luz gerada pelos reflexos difusos e especulares de outras superfícies e também pela difusão da luz na atmosfera iluminam estas áreas.
 - O efeito atmosférico é fácil de se comprovar ao se comparar a característica de negritude quase absoluta das sombras mostradas em fotos da Lua, onde não há atmosfera para difundir a luz, com fotos de um ambiente com alta taxa de umidade atmosférica, que difunde a luz, como a floresta amazônica.



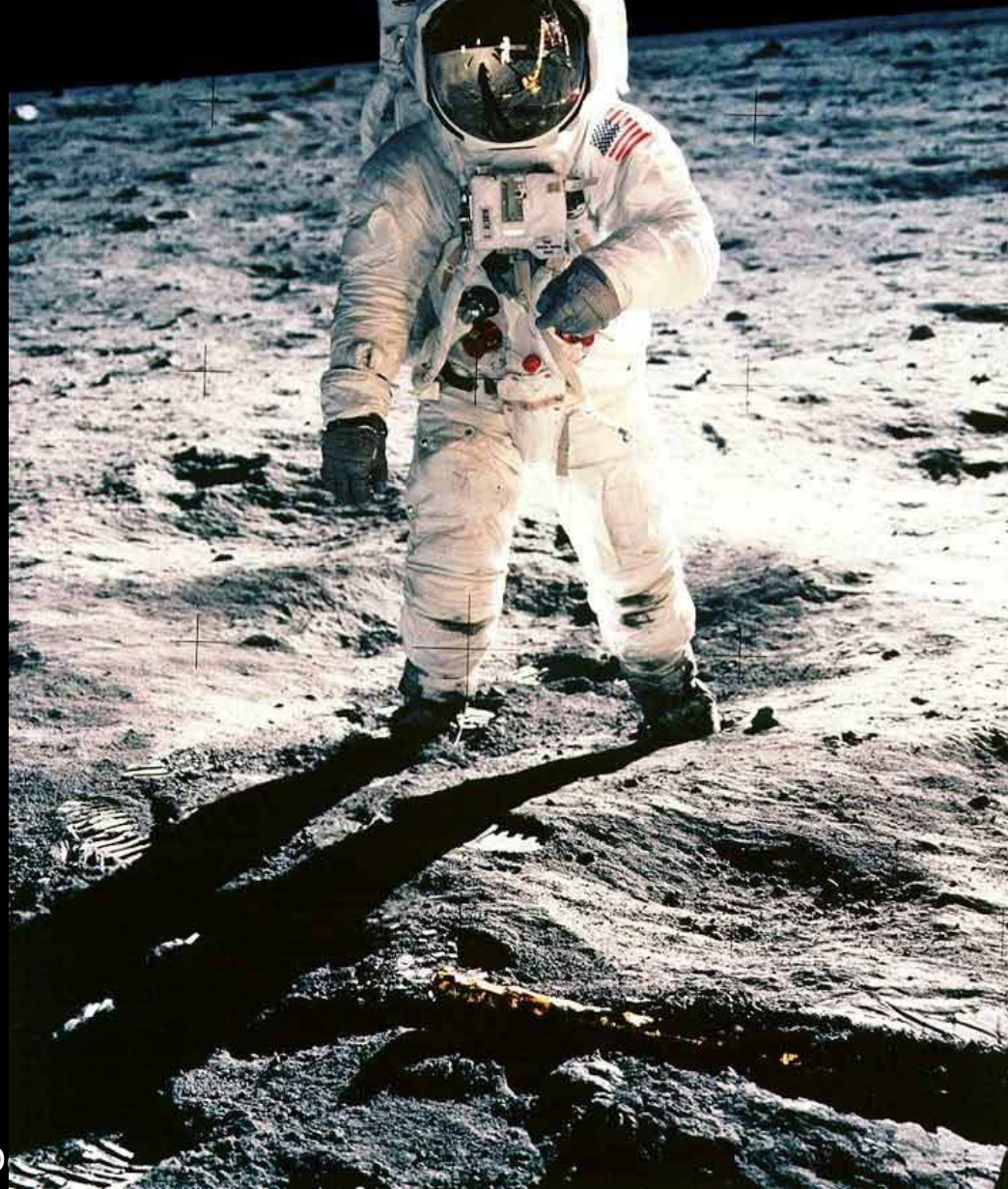
The Cyclops Project

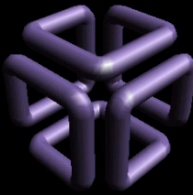
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Luz Ambiente

Em ambientes
sem atmosfera as
sombras são
duras.

Para simular a
dispersão da luz
causada pela
atmosfera,
necessitamos da
luz ambiente.





Luz Ambiente

- Calcular estas interações entre objetos e das fontes de luz e dos objetos com a atmosfera é extremamente complexo e se chama **método da radiância**.
- A não ser que estejamos desenvolvendo uma aplicação onde isto é de suma importância, podemos utilizar um **enfoque simplificado** para tratar este problema:
 - Consideramos a quantidade de luz que queremos que este objeto receba de seu ambiente como se fosse uma luz emitida pelo próprio objeto.
 - Para modelar isto, o modelo de Phong utiliza um termo de iluminação ambiente constante que, quando multiplicado pela coeficiente de reflectividade ambiente, nos dá a componente de luz ambiente da iluminação deste objeto.



Luz Ambiente - Phong

$$I_{amb} = I_A r_{amb}$$

I_{amb} - Intensidade da Luz Ambiente na Superfície

I_A - Intensidade Global de Luz Ambiente

r_{amb} - Coef. De Reflectividade Ambiente da Superfície

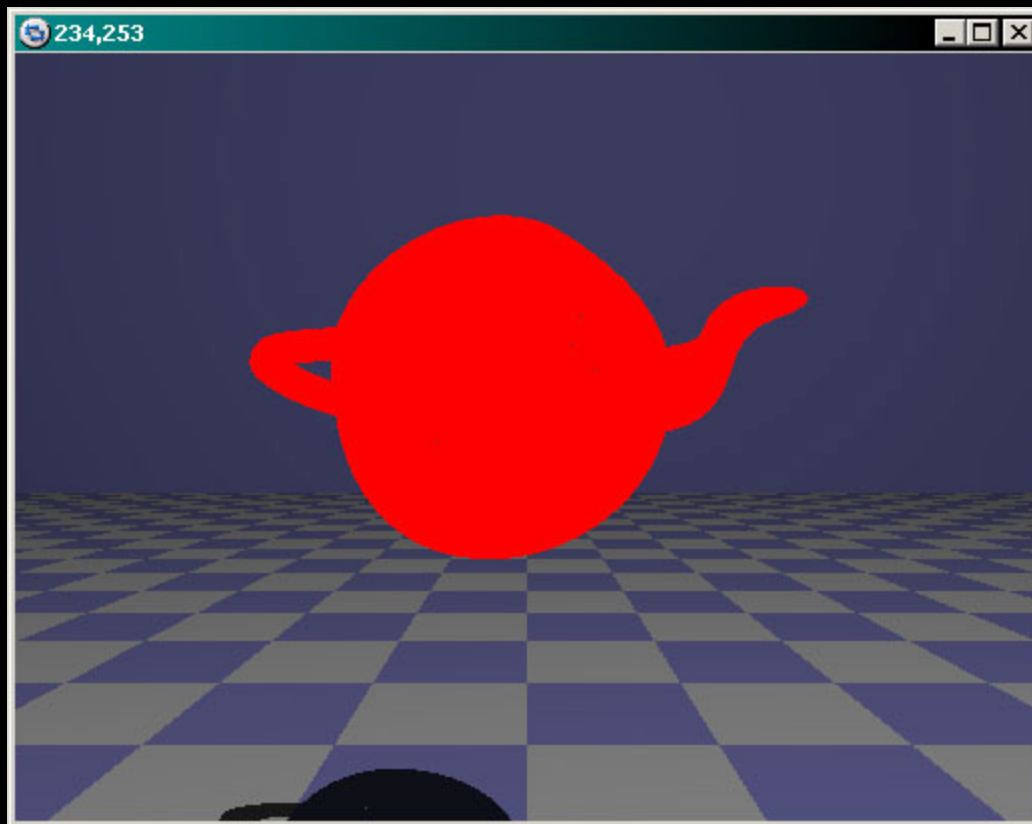


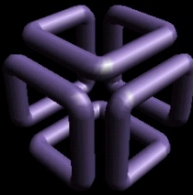
The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC





Integrando as Componentes - Phong

$$I = I_{dif} + I_{espec} + I_{amb}$$

$$I = I_L r_{dif}(\vec{v}_L \cdot \vec{v}_n) + I_L r_{espec}(\vec{v}_{ref} \cdot \vec{v}_{obs})^s + I_A r_{amb}$$



Integrando as Componentes – Phong

Acrescentando atenuação pela distância:

$$I = I_{amb} + \frac{(I_{espec} + I_{dif})}{(d + k)}$$

d - Distância do Emissor ao Ponto da Superfície

k - Parâmetro para aumentar ou diminuir o efeito da distância

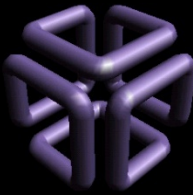


The Cyclops Project

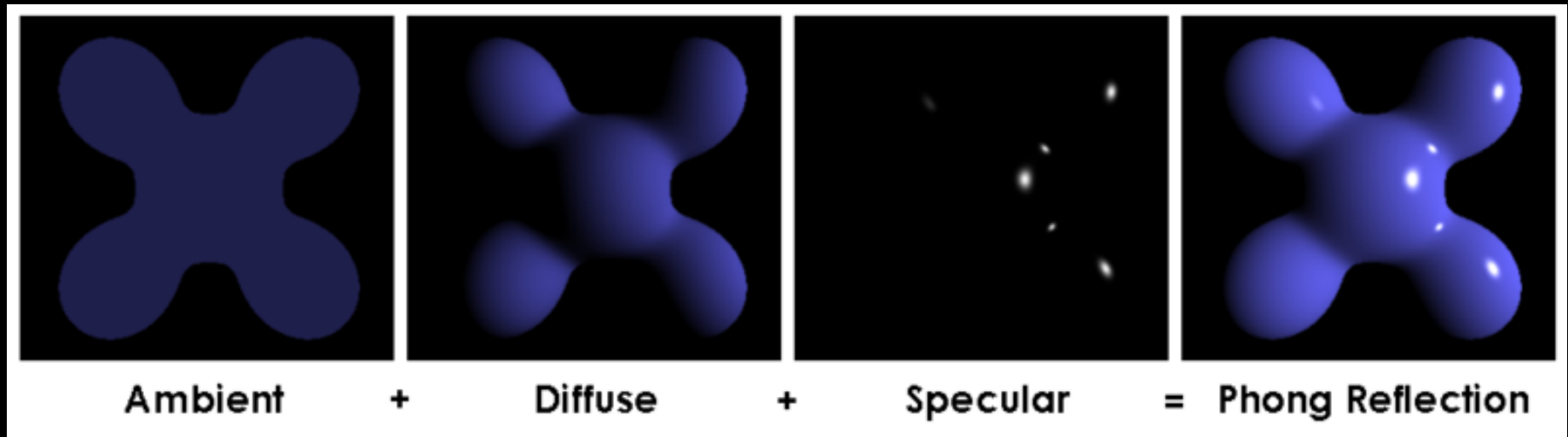
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

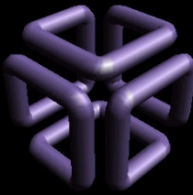
Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



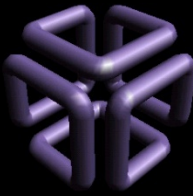
Integrando as Componentes – Phong





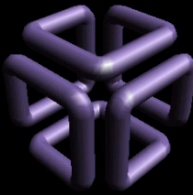
Conteúdo desta Aula

- Modelo de Iluminação de Phong
 - Luz Difusa
 - Luz Especular
 - Luz Ambiente
 - Equação de Phong
- Técnicas de Iluminação
 - Raytracing
 - Raycasting
 - Rayshading
 - Pixel Shading

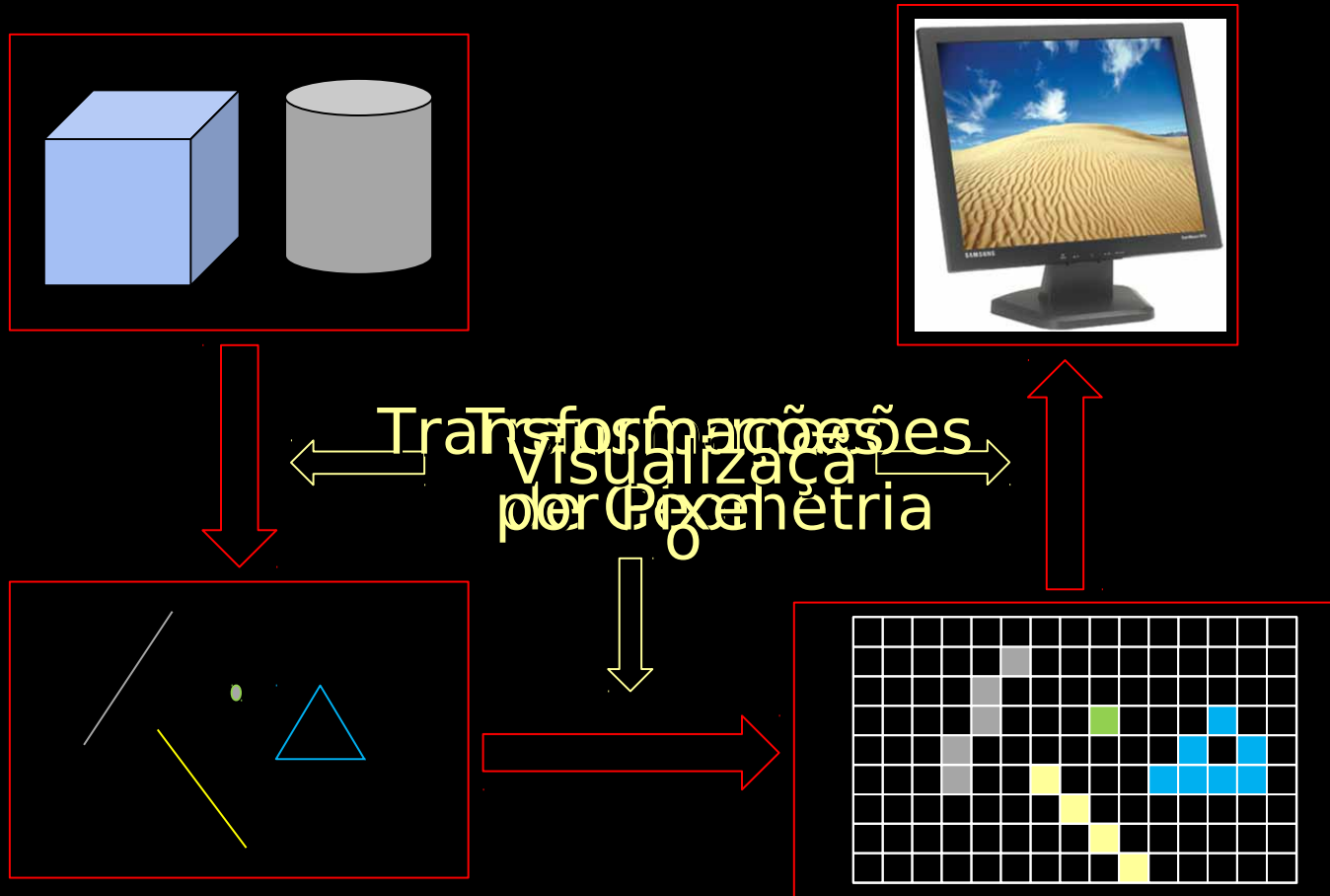


Técnicas de Iluminação

- “Raytracing” é um termo geral usado para indicar que a renderização de uma cena inclui iluminação.
- Existem várias maneiras de aplicar esta iluminação. Tudo depende dos requisitos da aplicação (em termos de velocidade processamento, fidelidade e realismo, etc) e da capacidade do hardware.
- Uma das características que distingue as técnicas de iluminação é o ponto onde o cálculo das luzes ocorre no *Pipeline de Renderização*.



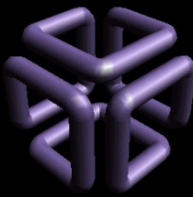
Pipeline de Renderização





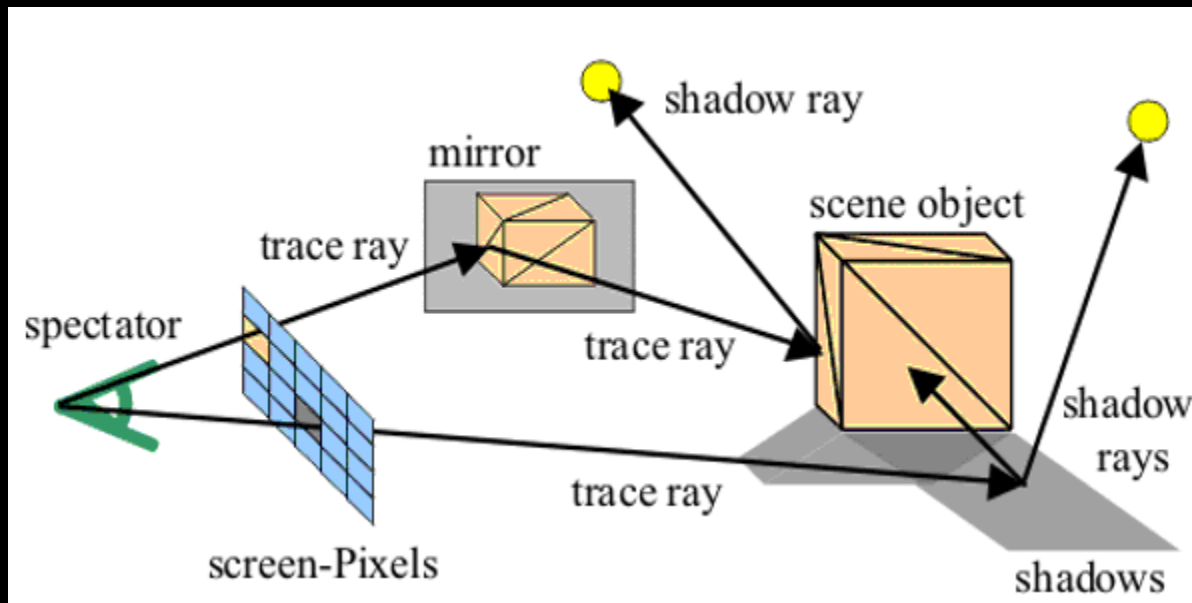
Etapa de Geometria

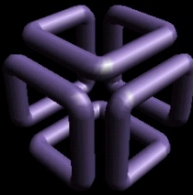
- Nesta etapa ocorrem as transformações nos objetos do mundo para prepará-los para renderização.
- São as operações “por vértice”.
- Os objetos são transformados para coordenadas da Window por PPC (é a Transformada de Visada ou *Viewing Transform*).
- Os objetos são projetados na Window pela Transformada de Projeção (*Projection Transform*).
- Os objetos são clipados para se considerar apenas a área de interesse.



Luz na etapa de Geometria

- A maioria das APIs gráficas implementa a iluminação dos objetos na etapa de Geometria.
 - É a etapa onde ainda se tem todas as informações de todos os objetos do mundo.
 - É o Raytracing clássico.





Luz na etapa de Geometria

- Ainda assim, existem diferentes maneiras de utilizar a idéia por trás de Raytracing para simular iluminação na etapa de Geometria:
- É possível traçar quantos raios existirem na resolução da Viewport, e seguir estes raios conforme estes colidem com os objetos do mundo até que percam intensidade suficiente.
 - Assim é possível gerar efeitos de sombra e superfícies reflexivas como espelhos.
 - O custo computacional é elevado e atualmente proibitivo para aplicações interativas.
 - É a premissa por trás de Raytracers estáticos como o POV-Ray.



Luz na etapa de Geometria

- Para aliviar o custo de processamento do Raytracing completo, APIs gráficas incluem algumas restrições ou “limitações” ao modelo.
- Tanto em OpenGL quanto em Direct3D, por exemplo, os raios que refletem em superfícies não são seguidos, eliminando-se assim a componente especular indireta.
- Isto limita a iluminação aos componentes de luz ambiente e direta, e efetivamente impossibilita efeitos como sombreamento sem intervenção do programador.
- Este “Raytracing simplificado” recebe o nome de Raycasting, que é mais eficiente e menos realista.



Luz na etapa de Geometria

- Finalmente, APIs como OpenGL e DirectX não traçam tantos raios quantos forem os pixels da Viewport.
- Ao invés disso, os raios são traçados apenas aos vértices dos polígonos dos objetos.
- A luz é calculada por Phong nos vértices, e então interpolada ao longo do resto do polígono.



The Cyclops Project

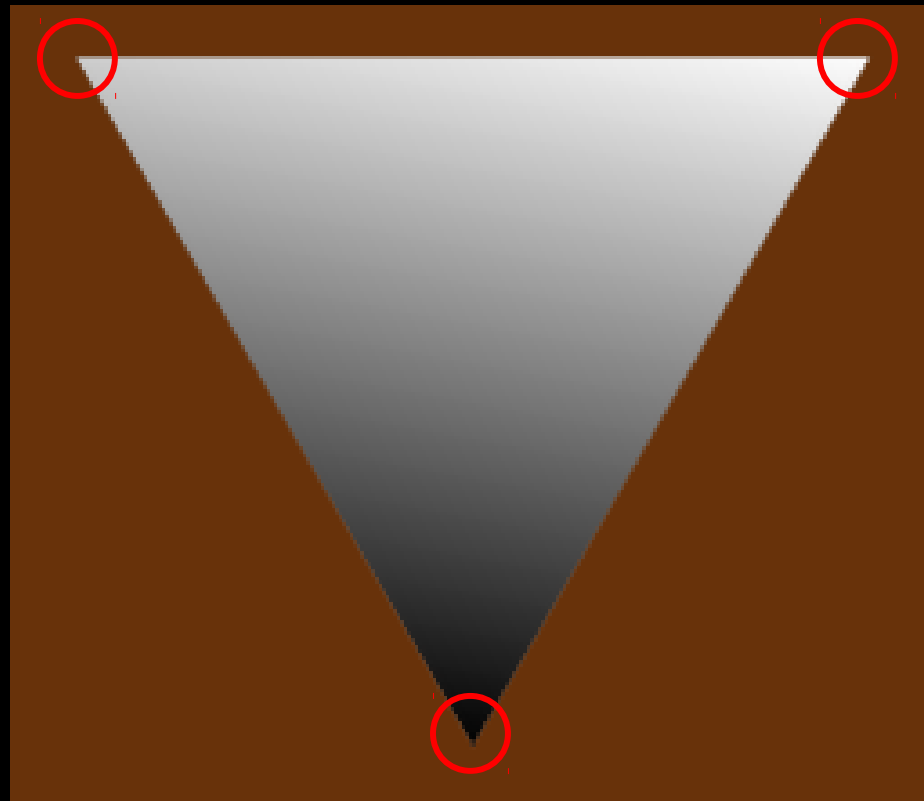
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



Luz na etapa de Geometria



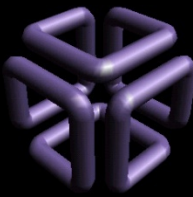


The Cyclops Project

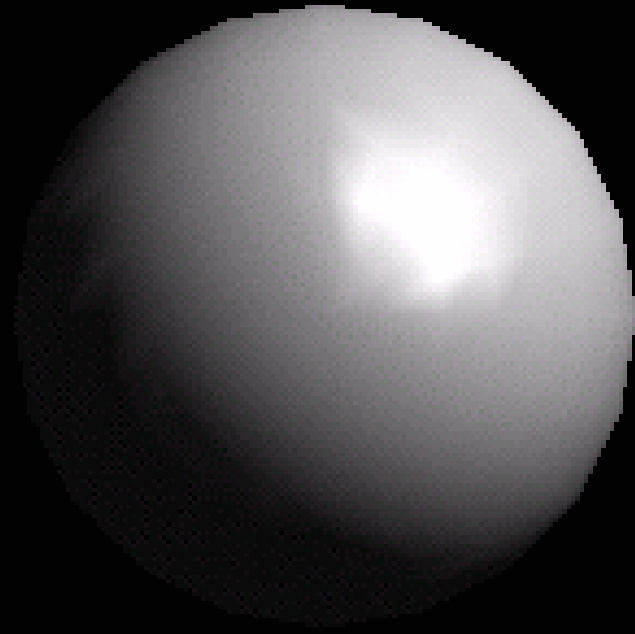
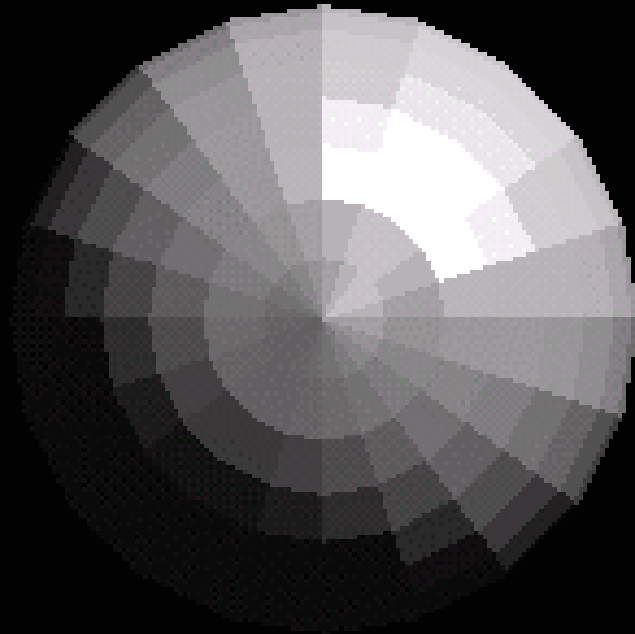
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

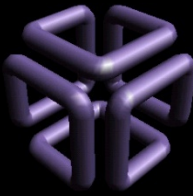
Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



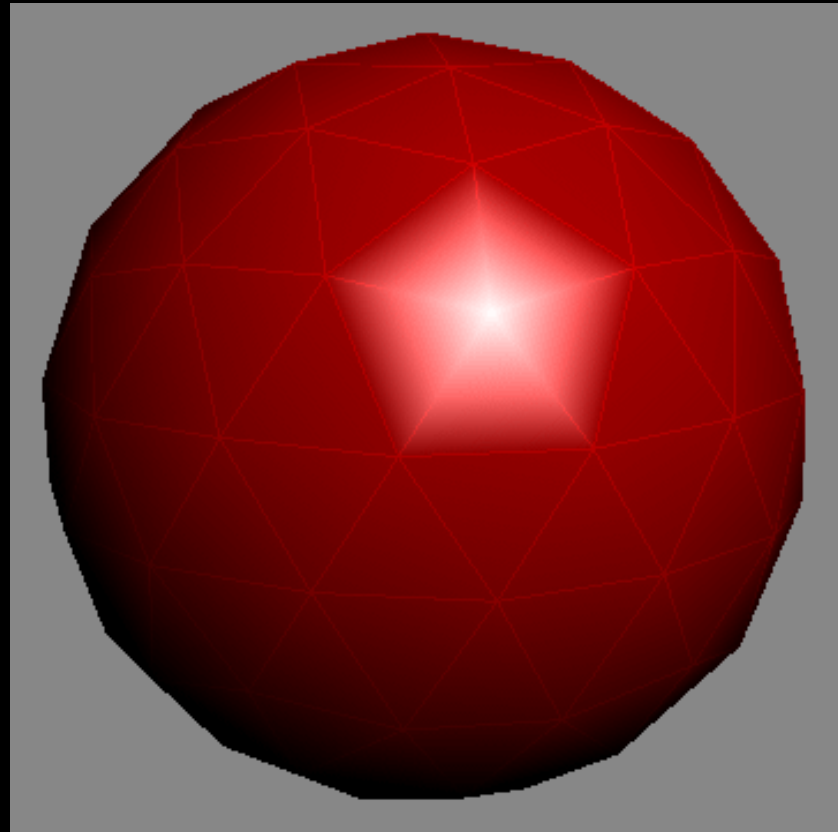
Flat Shading x Gouraud Shading





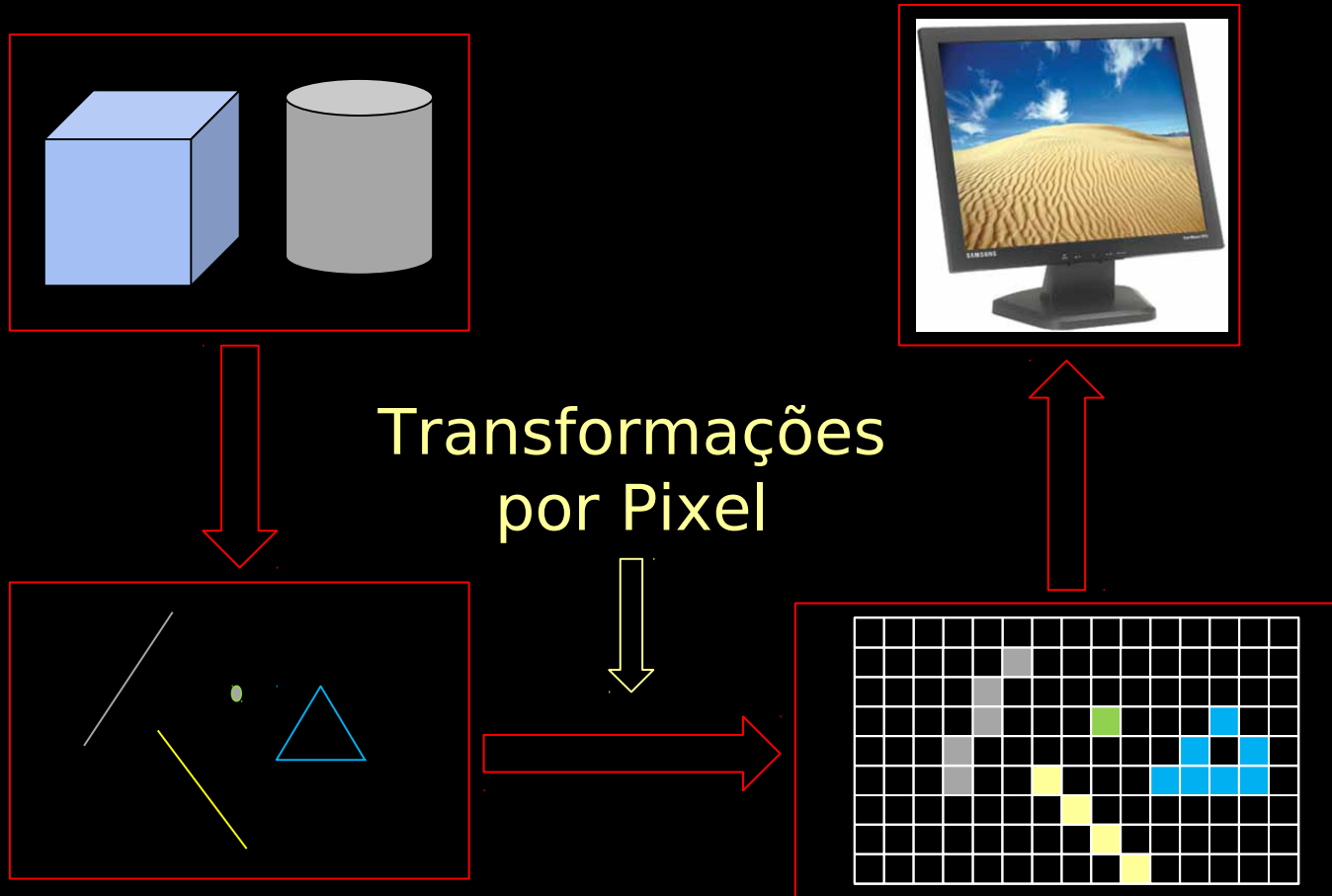
Flat Shading x Gouraud Shading

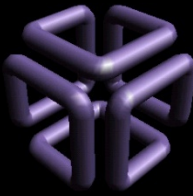
O número de
polígonos é
importante:
use poucos e
fica ruim...





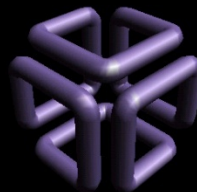
Pipeline de Renderização





Transformações por Pixel

- Nesta etapa ocorrem as transformações nas primitivas já descritas em coordenadas da Window e devidamente clipadas.
- A tarefa que falta é discretizar cada primitiva (reta, ponto, triângulo, etc) nos pixels que ela preencherá na Viewport, e colorir estes pixels.
- A profundidade de cada primitiva em relação à Window é considerada de forma que uma primitiva totalmente ocultada por outra não seja renderizada.
- O processo de transformar os objetos do mundo 3D em uma matriz de pixels é conhecido como Rasterização.



Luz na Etapa de Rasterização

- A noção de se considerar a iluminação de cada pixel individualmente (ao contrário de iluminar apenas os vértices dos objetos e interpolar o resultado) costumava ser impraticável pelo grande custo computacional.
- Apesar de mais barata que o Raytracing completo, a idéia de iluminar cada pixel (conhecida como Rayshading ou Pixel Shading) sofre a desvantagem de que na etapa de rasterização muita informação dos objetos já foi descartada.
- Por exemplo, é muito mais difícil gerar sombras com Pixel Shading.



Luz na Etapa de Rasterização

- Ainda assim, o Pixel Shading está começando a ser visto como um compromisso atraente entre o realismo extremo do Raytracing e a eficiência computacional do Raycasting.
- Ao iluminar cada pixel considerando sua normal, ganhamos o realismo que o Raycasting perde em interpolações.
- E como continuamos impondo a restrição de que os raios refletidos não são seguidos, ganhamos a velocidade de processamento que o Raytracing sacrifica.

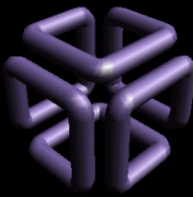


The Cyclops Project

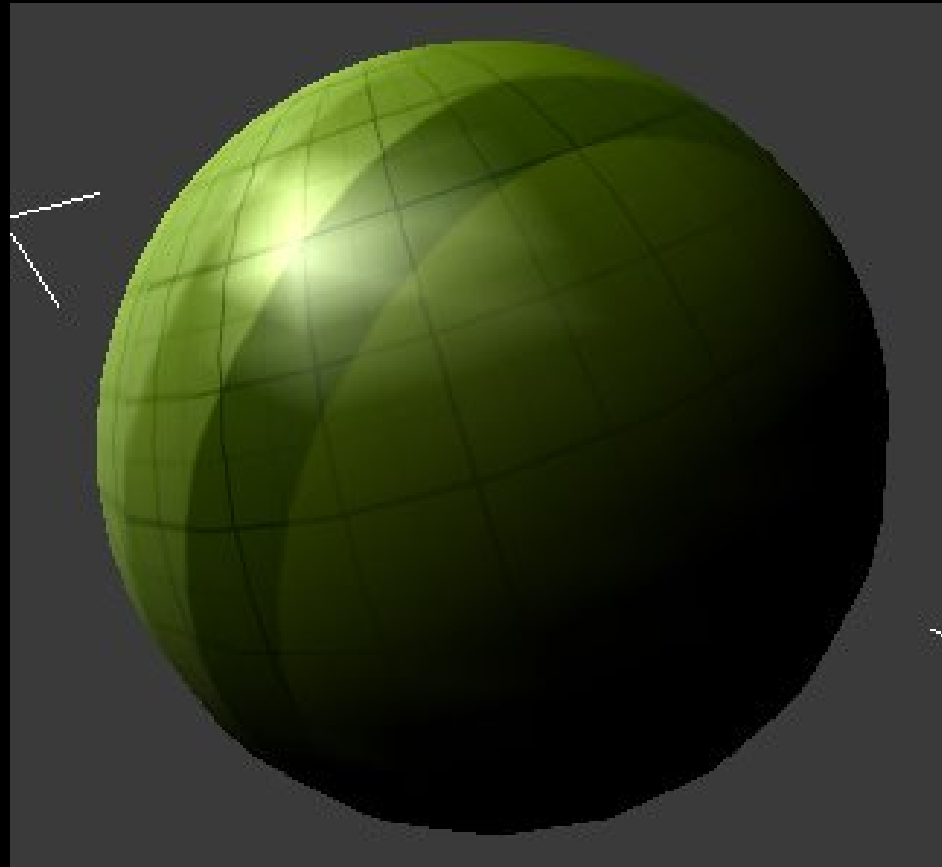
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



Luz na Etapa de Geometria (Gouraud Shading)





The Cyclops Project

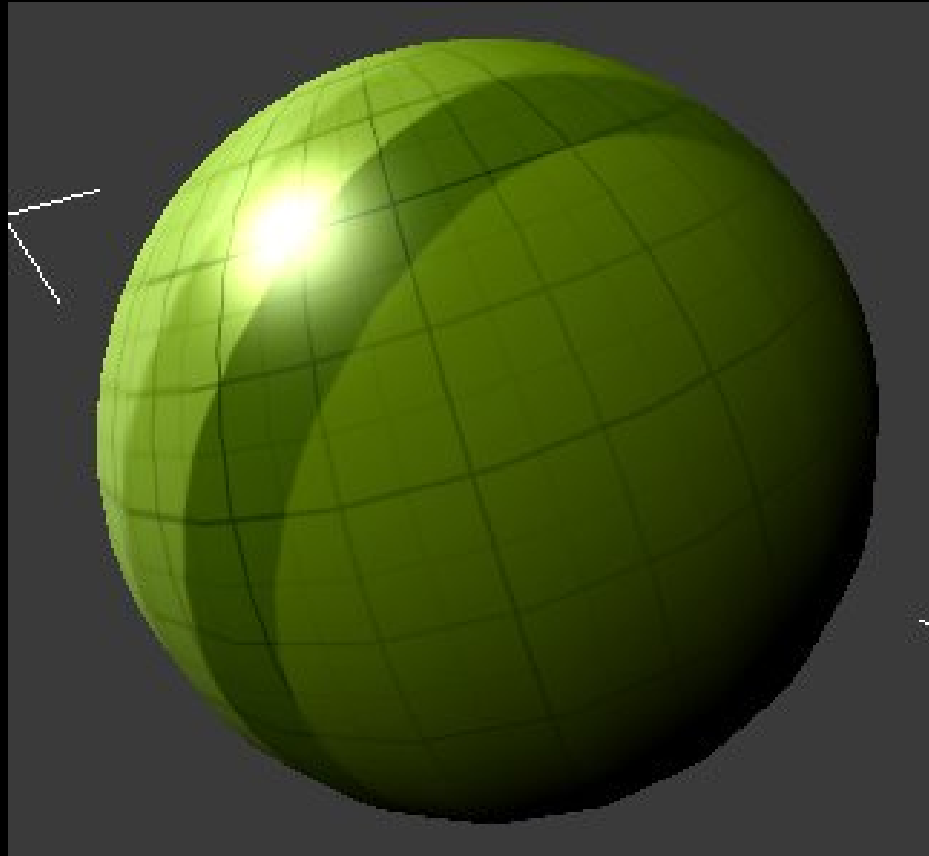
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



Luz na Etapa de Rasterização



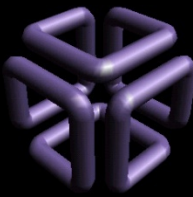


The Cyclops Project

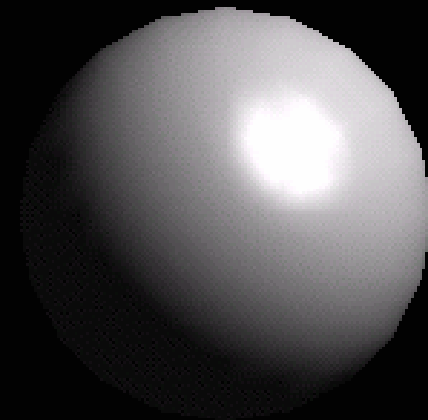
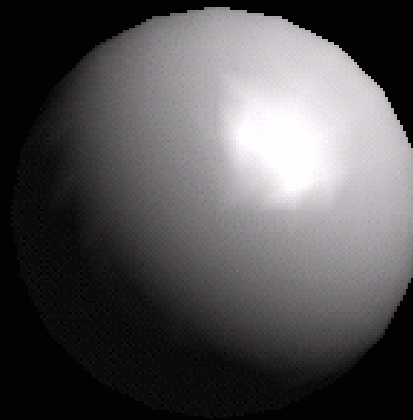
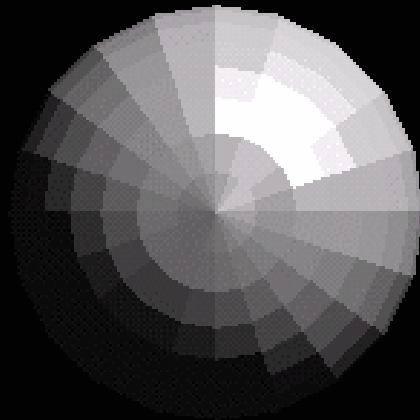
German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

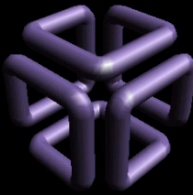
Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC



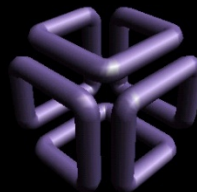
Flat Shading x Gouraud Shading x Pixel Shading





Luz no SGI

- No nosso Sistema Gráfico Interativo, a luz vai ser modelada através de Pixel Shading.
- Em poucas palavras, o acréscimo de funcionalidades no SGI será de aplicar a equação de Phong a *cada pixel* da Viewport.
- Isto significa acrescentar algumas estruturas de dados e algoritmos ao nosso Pipeline de Renderização.
- Desta forma, o novo Pipeline ao final do processo ficará assim:



Passos para implementação do Raytracing no SGI:

1. Criação da Window com uma orientação arbitrária
2. Transformação dos objetos do Mundo para PPC
3. Transformação de Projeção
4. Clipping *
5. Conversão por Varredura
6. Z-Buffering, complementarmente à Conversão por Varredura
7. Raytracing por Pixel (Pixel Shading)
8. Cópia do Z-Buffer para a Viewport



The Cyclops Project

German-Brazilian Cooperation Programme on IT
CNPq GMD DLR

Disciplina Computação Gráfica

Curso de Ciência da Computação
INE/CTC/UFSC

