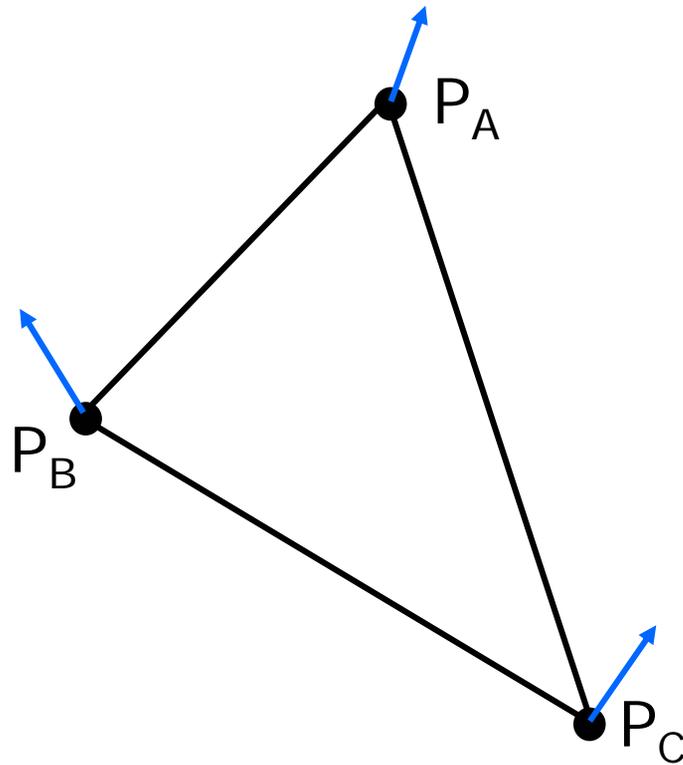


Kapitel18: Rasterung von Flächen

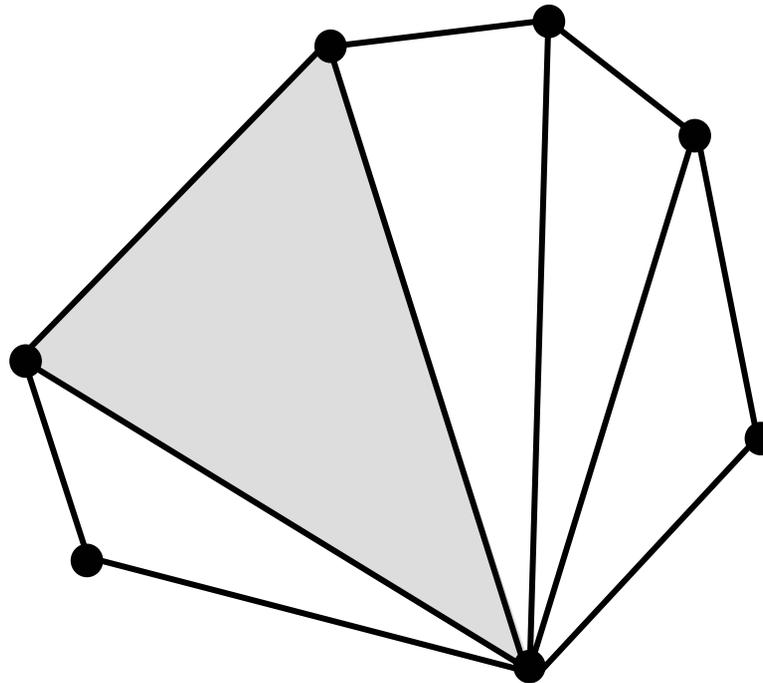
Ausgangslage

am Ende der Viewing Pipeline liegt vor:



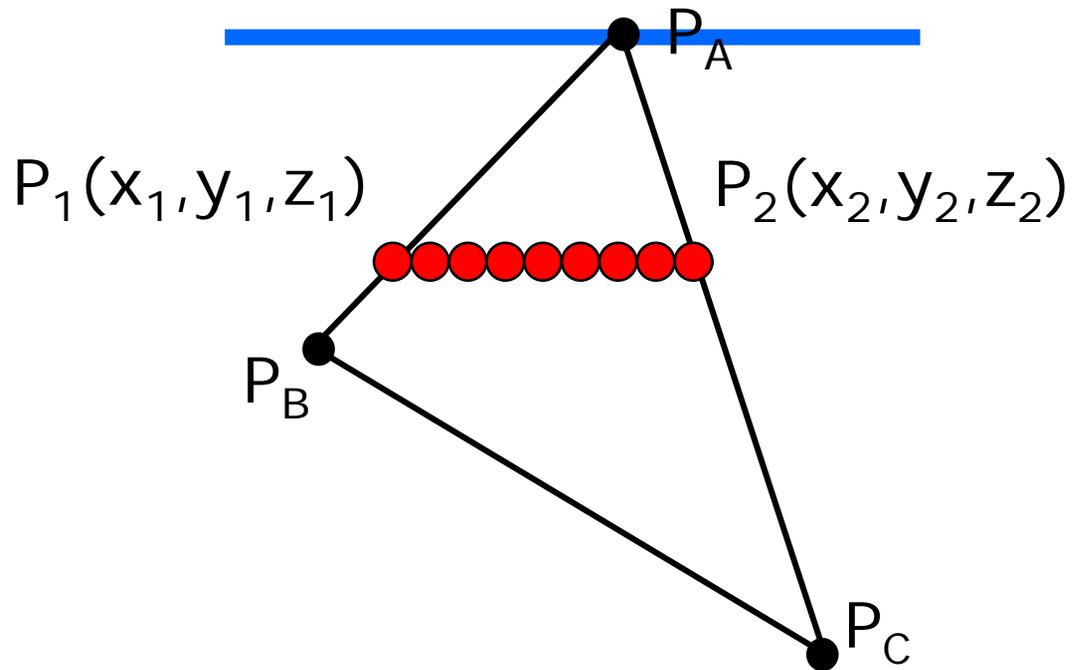
Triangulierung

konvexe Polygone triangulieren



Ziel

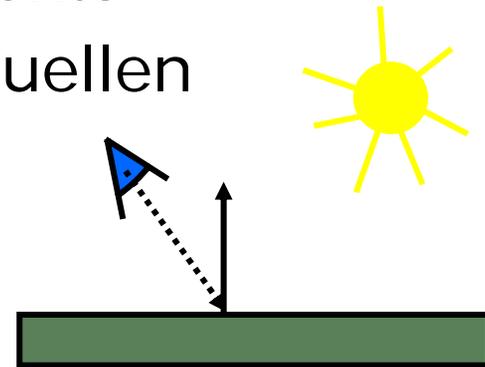
zeilenweises Rastern



Beleuchtung

zur Berechnung der Farbe in einem Punkt geht ein:

- Materialeigenschaften des Objekts
- Augenpunkt des Betrachters
- Normalenvektor des Objekts
- Positionierung der Lichtquellen
- Art der Lichtquellen



Lichtquellen

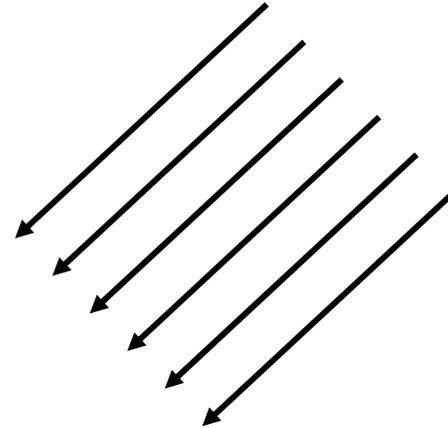
- Umgebungslicht ambient light
- gerichtetes Licht directed light
- Punktlicht point light
- Strahler spot light

Umgebungslicht

- keine Position
- keine Richtung
- Intensität I_a

Gerichtetes Licht

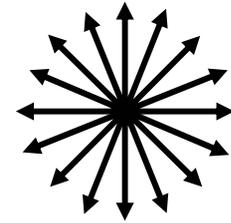
- keine Position
- Lichtrichtung L_d
- Intensität I_d



z.B. Sonnenlicht

Punktlicht

- Position L_p
- keine bevorzugte Richtung
- Anfangs-Intensität I_0
- Intensität nimmt mit Entfernung ab



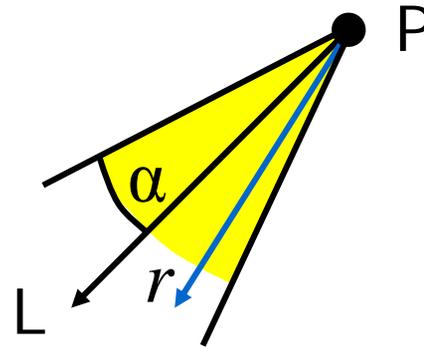
$$I(r) = \frac{I_0}{C_1 + C_2 \cdot r}.$$

$r \geq 0$ Abstand zur Lichtquelle

$C_1 \geq 1$ Abschwächungskoeffizienten

$C_2 \geq 0$

Strahler



- Position P
- Lichtrichtung L
- Intensität I_0
- Abschwächungskoeffizienten C_1, C_2
- [Abstrahlwinkel α]
- Konzentrationsexponent c
- Intensität im Lichtkegel $\cos(r, L)^c$

Reflexionseigenschaften

- ambientes Licht
- diffus reflektiertes gerichtetes Licht
- spekulär reflektiertes gerichtetes Licht

$$\bar{C} = \sum_{i=1}^n (\bar{C}_{a_i} + \bar{C}_{d_i} + \bar{C}_{s_i}) .$$

n Zahl der Lichtquellen
 \bar{C}_{a_i} ambienter Anteil von Lichtquelle i
 \bar{C}_{d_i} diffuser Anteil von Lichtquelle i
 \bar{C}_{s_i} spekulärer Anteil von Lichtquelle i
pro Farbe einzeln berechnen !

Oberflächeneigenschaften

k_a ambienter Reflexionskoeffizient

k_d diffuser Reflexionskoeffizient

k_s spekulärer Reflexionskoeffizient

$\overline{O_d}$ diffuse Objektfarbe

$\overline{O_s}$ spekulare Objektfarbe

O_e spekulärer Exponent

ambiante Reflexion

Grundhelligkeit eines Objekts

$$\overline{C}_a = k_a \cdot I_a \cdot \overline{O}_d$$

k_a ambienter Reflexionskoeffizient

I_a Intensität des ambienten Lichts

\overline{O}_d diffuse Objektfarbe

diffuse Reflexion

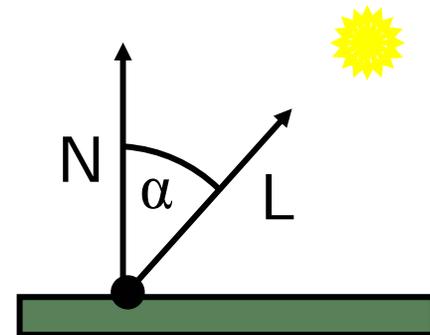
vom Objekt diffus reflektiertes Licht,
d.h. überall gleichmäßig sichtbar

$$\overline{C}_d = k_d \cdot I_e \cdot \overline{O}_d \cdot \cos(L, N)$$

k_d diffuser Reflexionskoeffizient

I_e Intensität des
einfallenden Lichts

\overline{O}_d diffuse Objektfarbe



spekulare Reflexion

vom Objekt gespiegeltes Licht,
nur in bestimmter Richtung sichtbar

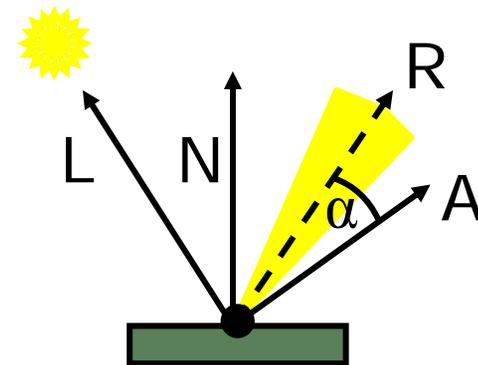
$$C_s = k_s \cdot I_e \cdot \overline{O_s} \cdot \cos(R, A)^{O_e}$$

k_s diffuser Reflexionskoeffizient

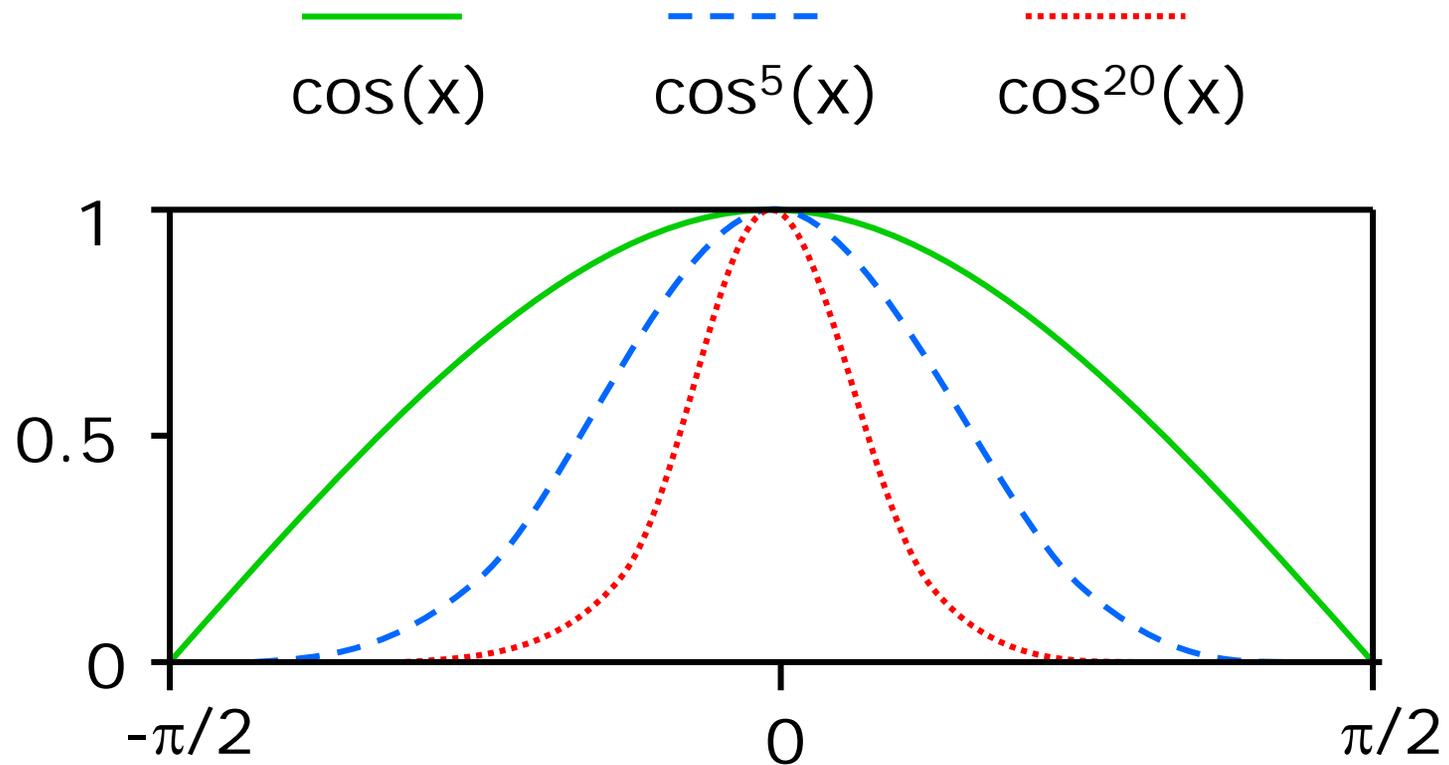
I_e Intensität des einfallenden Lichts

$\overline{O_s}$ spekulare Objektfarbe

O_e spekulärer Exponent,
regelt Streukegel



Intensität im Streukegel



Materialeigenschaften

nicht mehr abstrahlen
als empfangen:

$$0 \leq k_a, k_d, k_s \leq 1$$

bei gerichtetem Licht:

$$k_d + k_s \leq 1$$

kontrastarm

$$k_a \gg k_d, k_s$$

matt

$$k_d \gg k_s$$

spiegelnd

$$k_s > k_d$$

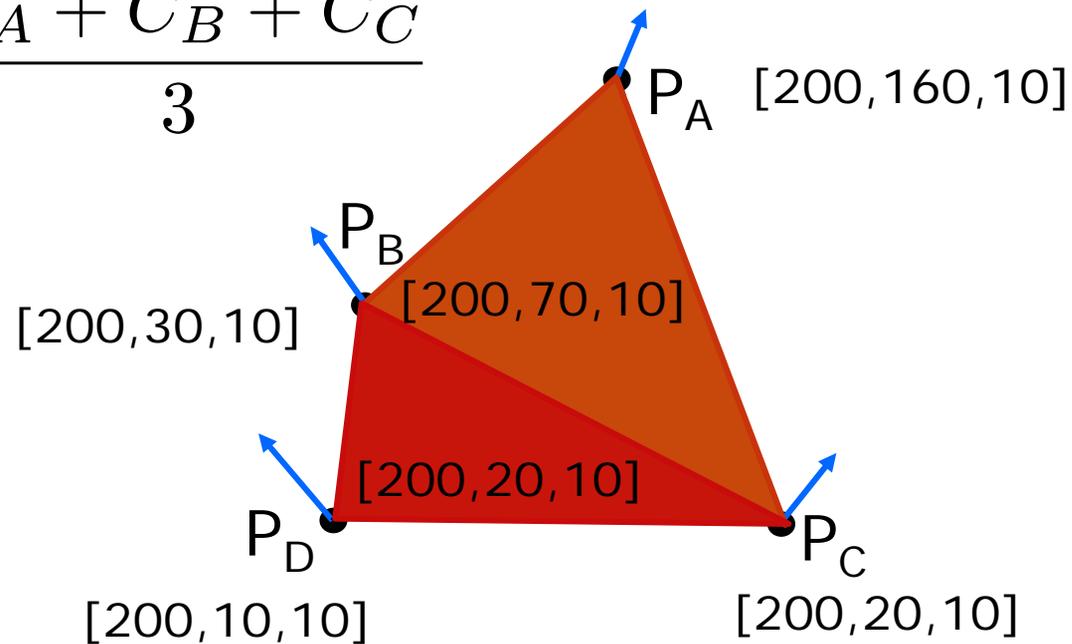
Schattierungsalgorithmen

- Flat Shading:
pro Dreieck eine Farbe
- Gouraud Shading:
Interpolation der Farbwerte
- Phong Shading:
Interpolation der Normalen

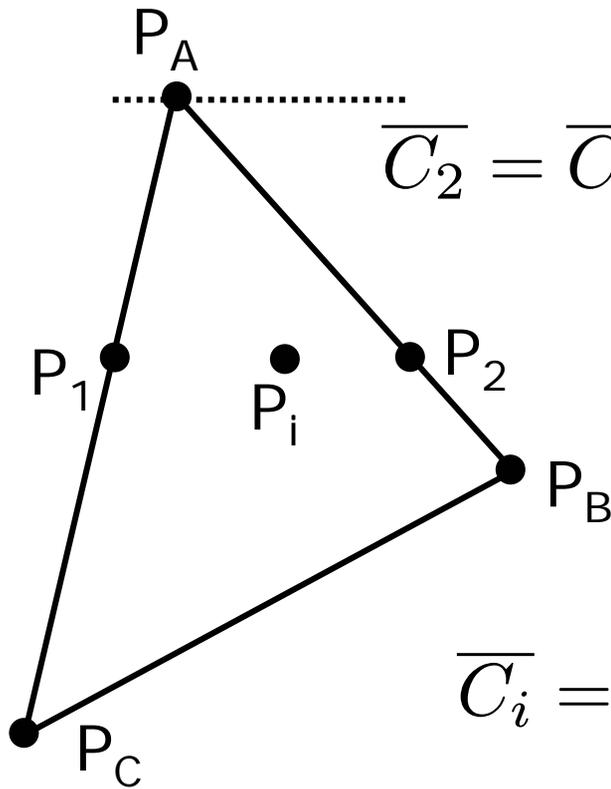
Flat Shading

- Eckpunkte im WC beleuchten
- Mittelwert für alle Pixel

$$\overline{C}_i = \frac{\overline{C}_A + \overline{C}_B + \overline{C}_C}{3}$$



Gouraud Shading

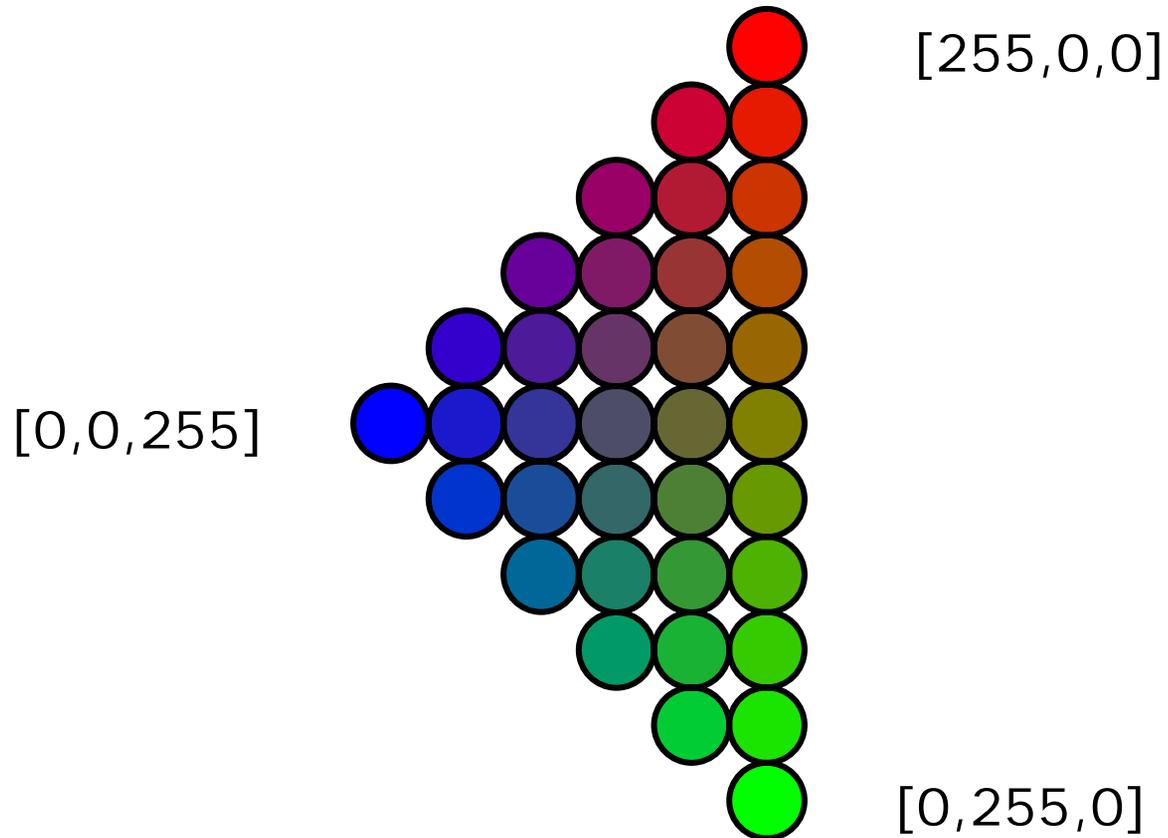


$$\overline{C}_1 = \overline{C}_A \frac{y - y_C}{y_A - y_C} + \overline{C}_C \frac{y_A - y}{y_A - y_C}$$

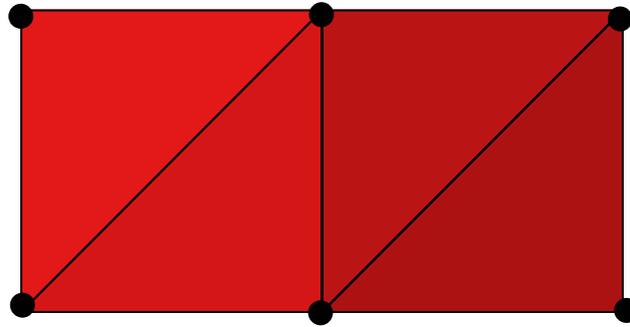
$$\overline{C}_2 = \overline{C}_A \frac{y - y_B}{y_A - y_B} + \overline{C}_B \frac{y_A - y}{y_A - y_B}$$

$$\overline{C}_i = \overline{C}_1 \frac{x_2 - x_i}{x_2 - x_1} + \overline{C}_2 \frac{x_i - x_1}{x_2 - x_1}$$

Gouraud Shading



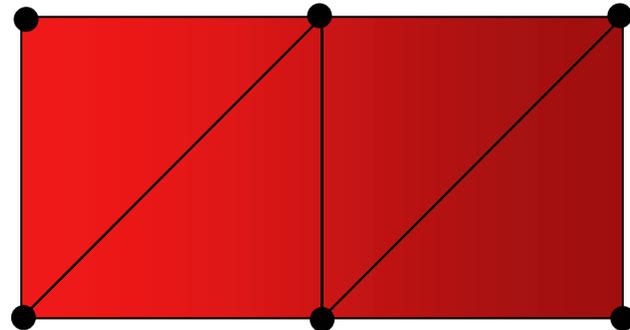
Flat versus Gouraud



[240,24,24]

[200,20,20]

[160,16,16]



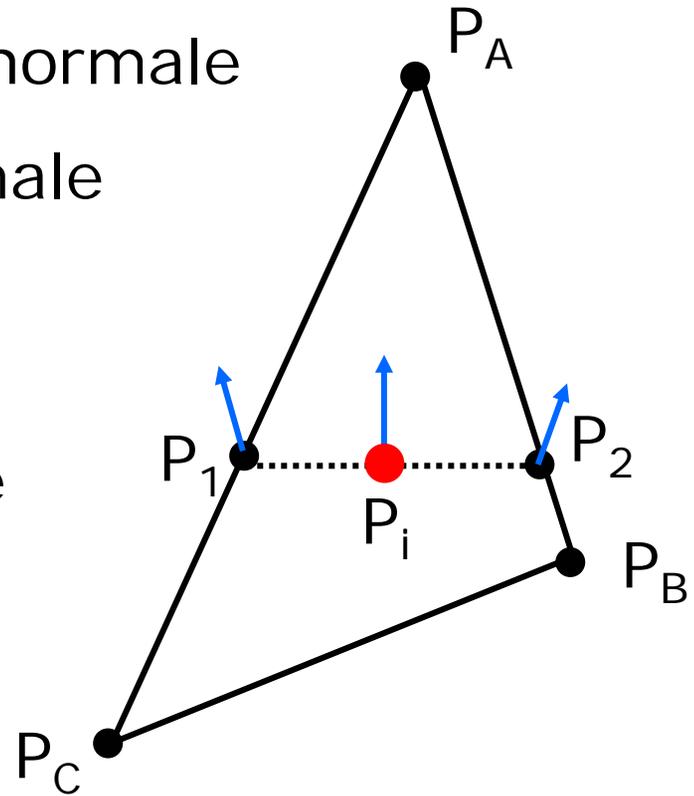
Phong Shading

pro Scanline:

- interpoliere Anfangsnormale
- interpoliere Endnormale

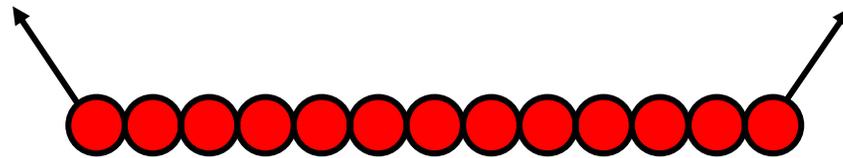
pro Pixel:

- interpoliere Normale
- berechne Farbwert

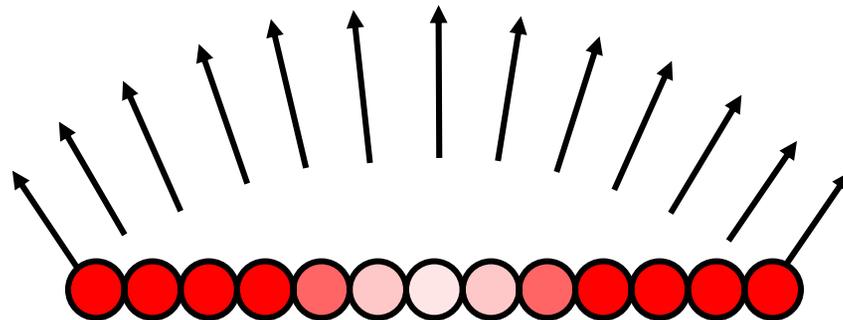


Gouraud versus Phong

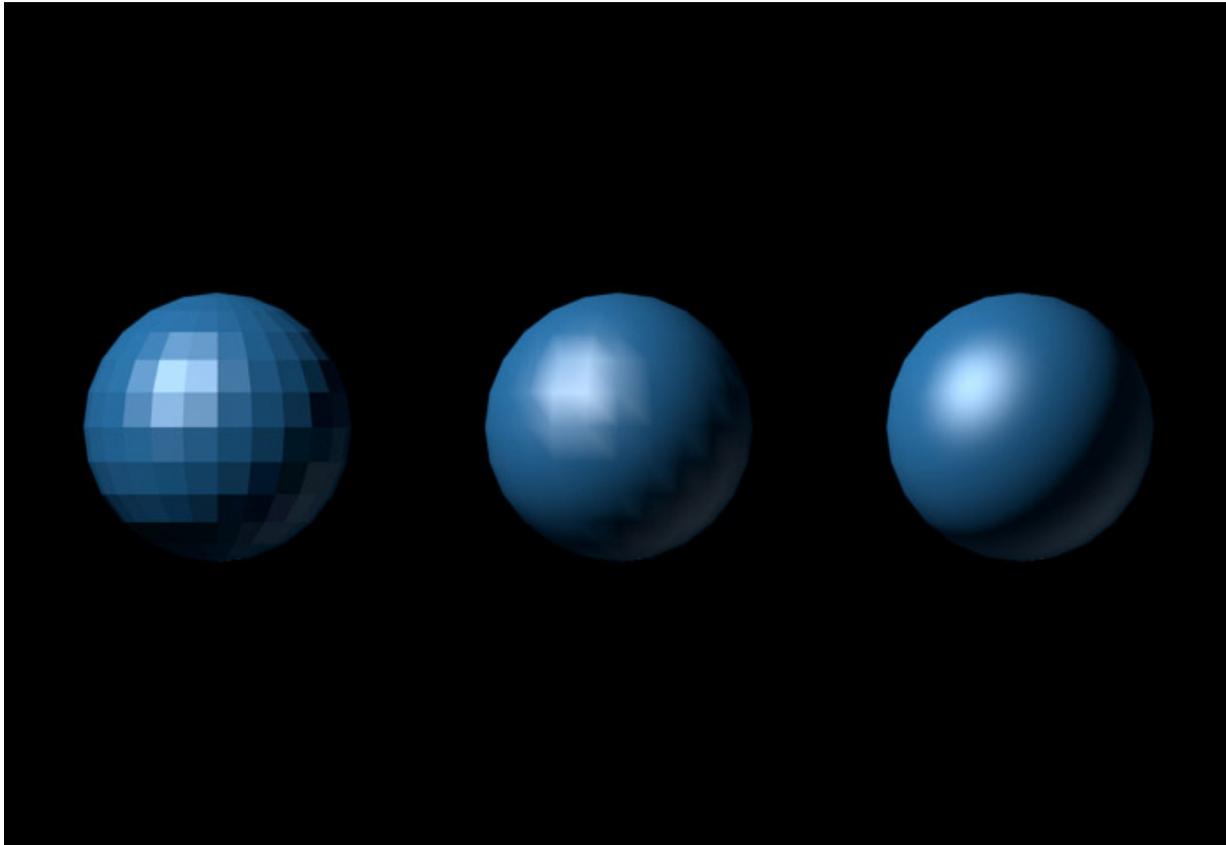
Farbwerte interpolieren



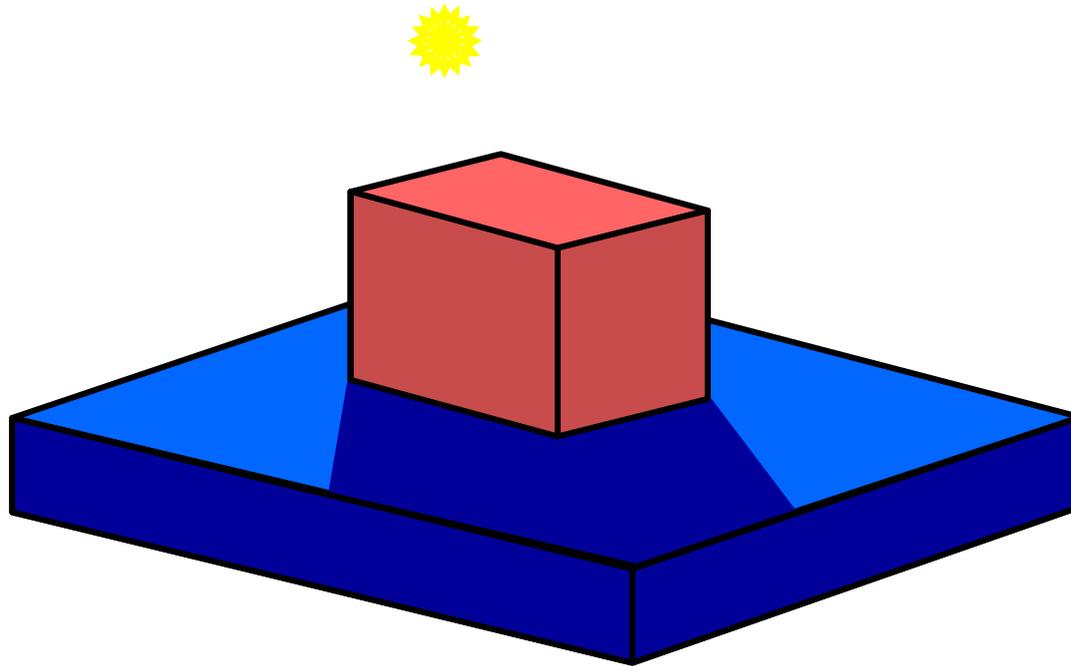
Normalen interpolieren
Farbwerte ausrechnen



Flat versus Gouraud versus Phong



Schatten



von der Lichtquelle nicht sichtbare Pixel

Berechnung von Schatten

geeignet: Hidden-Surface-Removal-Algorithmen

Phase 1: rendere Bild aus Position der Lichtquelle in einen Schattentiefenpuffer $s_tiefe[][]$

Phase 2: rendere Bild aus Position des Betrachters mit modifiziertem Tiefenpuffer-Algorithmus:

falls Pixel (x,y,z) sichtbar, transformiere P in den Koordinatenraum von Phase 1. Ergibt $P'(x',y',z')$.

Falls $z' \geq s_tiefe[x',y']$ dann P nicht im Schatten

Falls $z' < s_tiefe[x',y']$ dann P im Schatten

\Rightarrow Helligkeit reduzieren

Berechnung von Schatten

