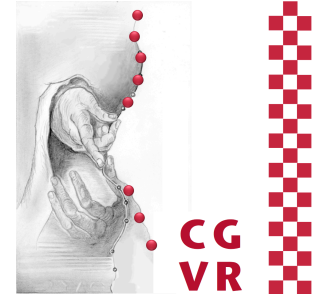
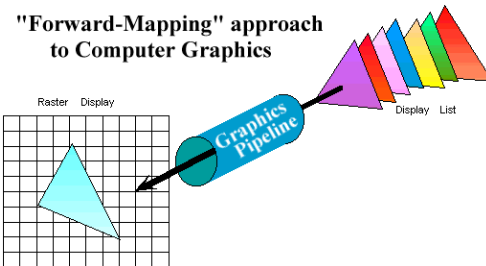


Bremen



Computergraphik I

Die Graphik-Pipeline



G. Zachmann

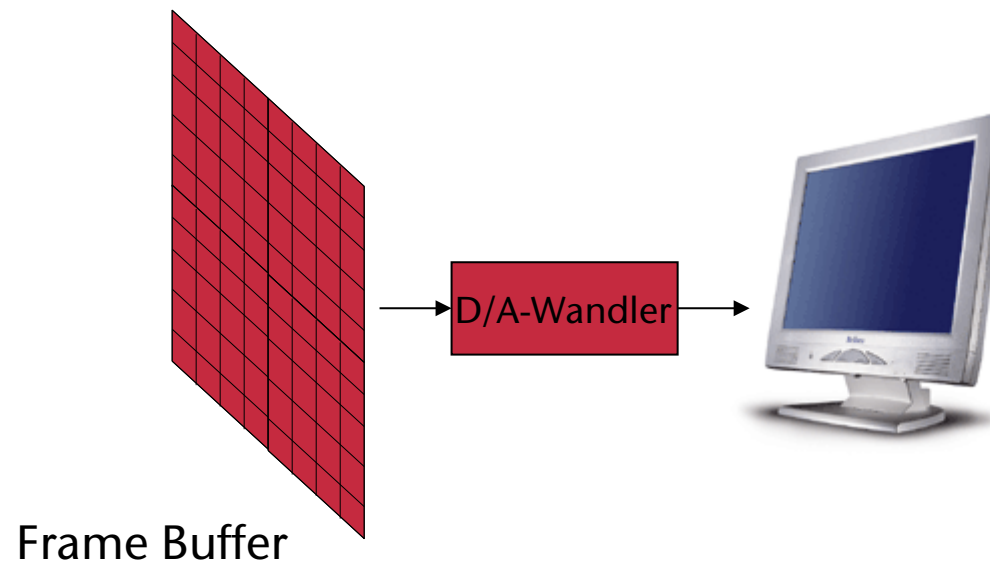
University of Bremen, Germany

cgvr.informatik.uni-bremen.de

- Alle Geometrie wird aus folgenden **graphischen Primitiven** zusammengesetzt:
 - Punkte
 - Einzelne Strecken (= "Geraden" oder "Linien")
 - Linienzug (= Folge von zusammenhängenden Linien)
 - Dreiecke (evtl. konvexe Polygone)
 - Buchstaben (vordefinierte Zeichen)
- Das sind alle!
 - Kurven? → werden aus einer Folge von Linien approximiert
 - Allgemeine Polygone? → werden i.A. in Dreiecke unterteilt
 - Gekrümmte Flächen? → werden i.A. durch Dreiecke approximiert
- Der Trend geht in Richtung einfacher Primitive
 - Einfach, einheitlich, sich oft wiederholend: gut für Parallel-Verarbeitung

Der Output: ein Array von Farbwerten

- Auf der Graphikkarte haben wir einen eigenen Bildspeicher
- Hier schreiben wir letzten Endes einzelne Pixel
- Ein Digital/Analog-Wandler liest diesen periodisch aus und wandelt den Inhalt in das Bildsignal für den Monitor um
- Typisches Format: 3 Bytes pro Pixel (r,g,b)



"A Trip Down the Graphics Pipeline"



■ Input:

- Geometric model: description of all objects, surfaces,
- Transformations
- Lighting model: Computational description of object and light properties, interaction (reflection)
- Synthetic Viewpoint (or Camera): Eye position and viewing frustum
- Raster Viewport (Frame Buffer): Pixel grid onto which image plane is mapped



Model-to-World Transformation

Illumination (Shading)

Viewing Transformation (Perspective / Orthographic)

Clipping

Projection (in Screen Space)

Scan Conversion (Rasterization)



Visibility / Display

■ Output:

- Colors / Intensities suitable for framebuffer display (for example, 24 Bits RGB values at each pixel)

Hier sind wir (Programmierer)



APPLICATION

COMMAND STREAM

3D Transformation, Shading



GEOMETRY PROCESSING

TRANSFORMED GEOMETRY

Konvertierung der Primitive in Pixel



RASTERIZATION

FRAGMENTS

Blending, Pixel Tests, (Shading)



FRAGMENT PROCESSING

FRAMEBUFFER IMAGE

Ausgabe auf dem Bildschirm

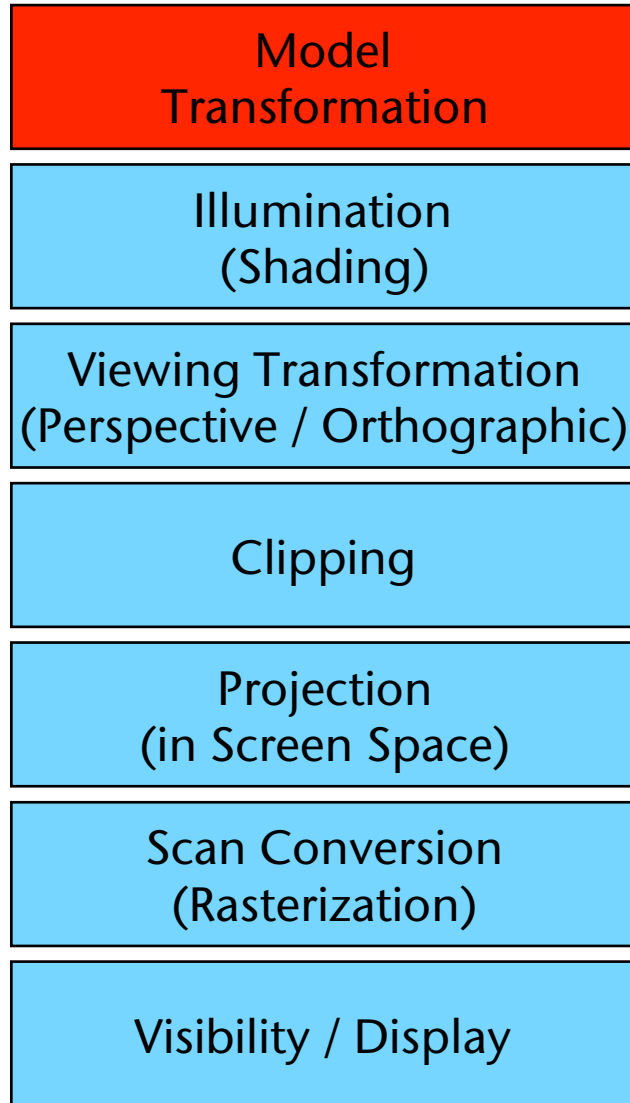
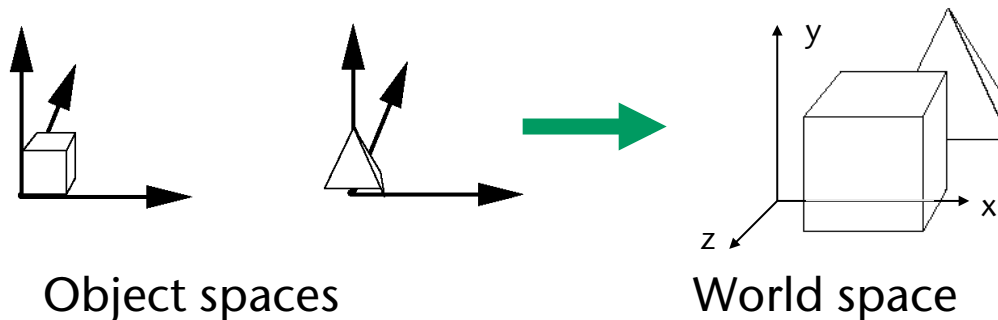


DISPLAY



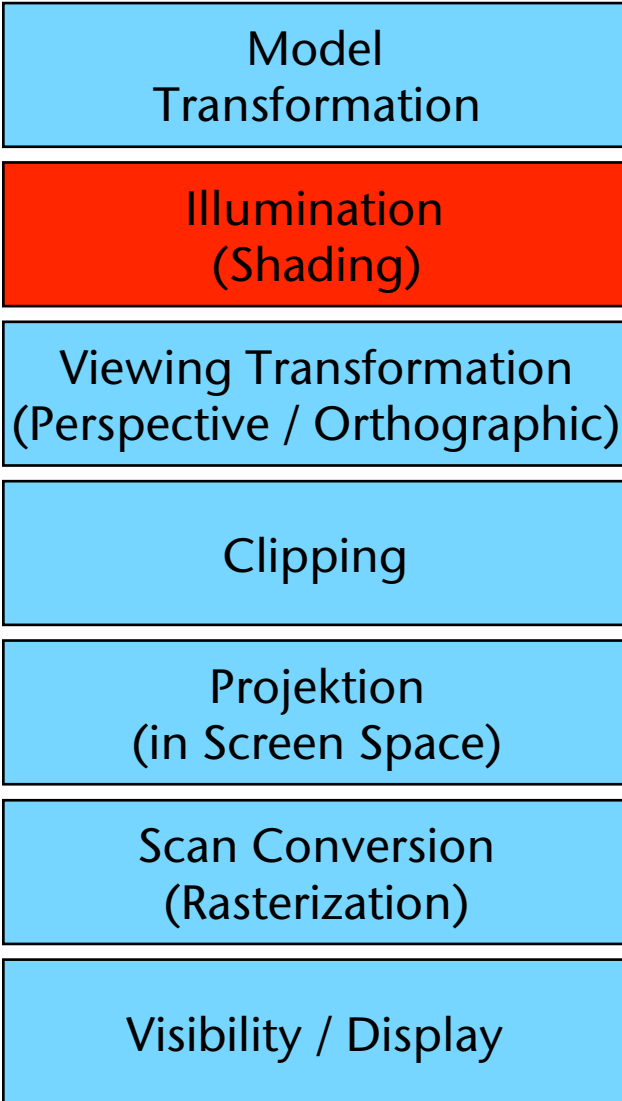
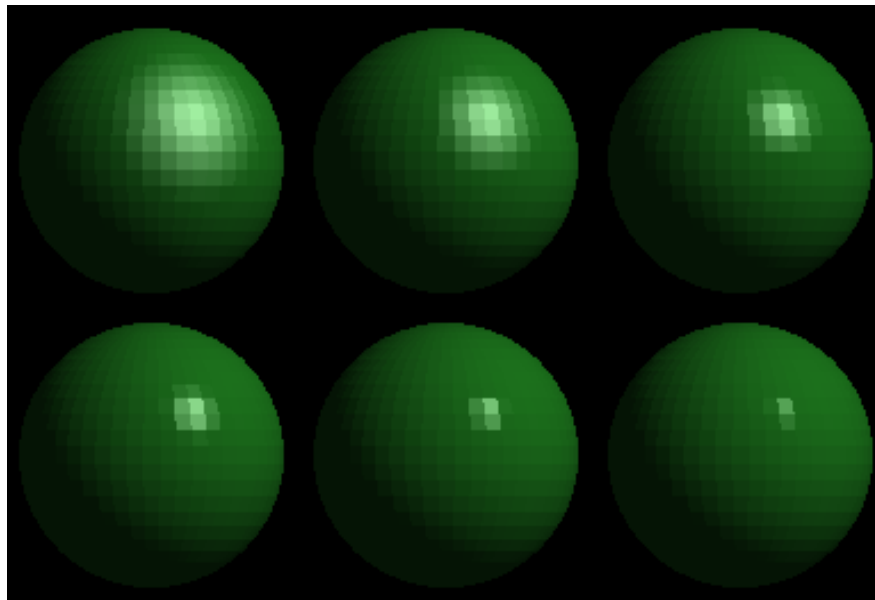
Model Transformation (model-to-world)

- 3D Modell wird im eigenen Koordinatensystem definiert (**object space**)
- Model Transformation** positioniert die Objekte in einem allg. Koordinatensystem (**world space**)



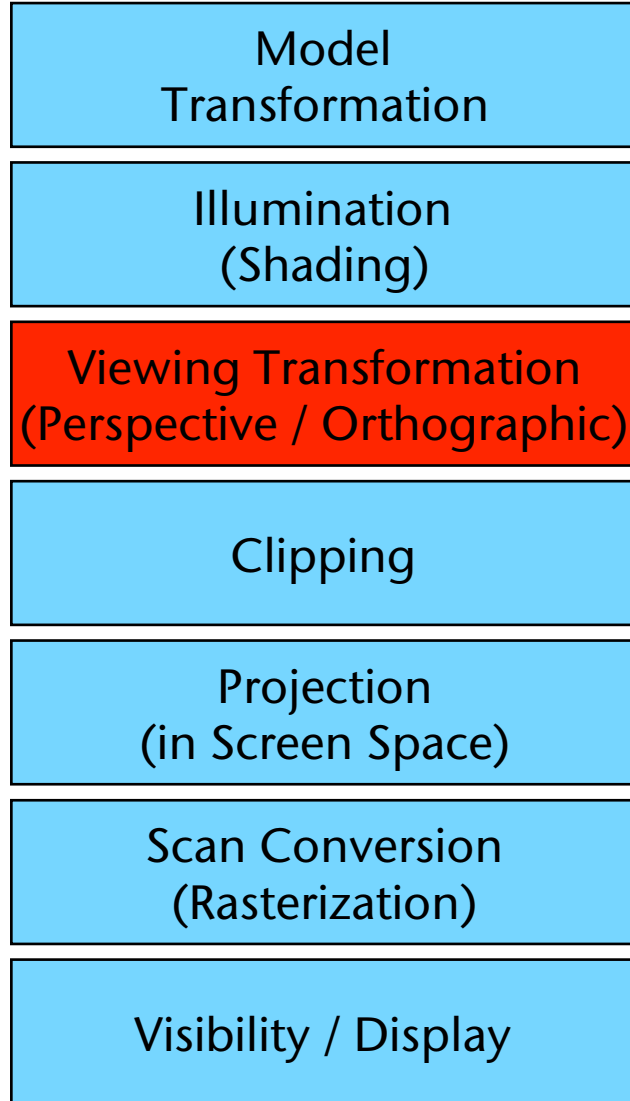
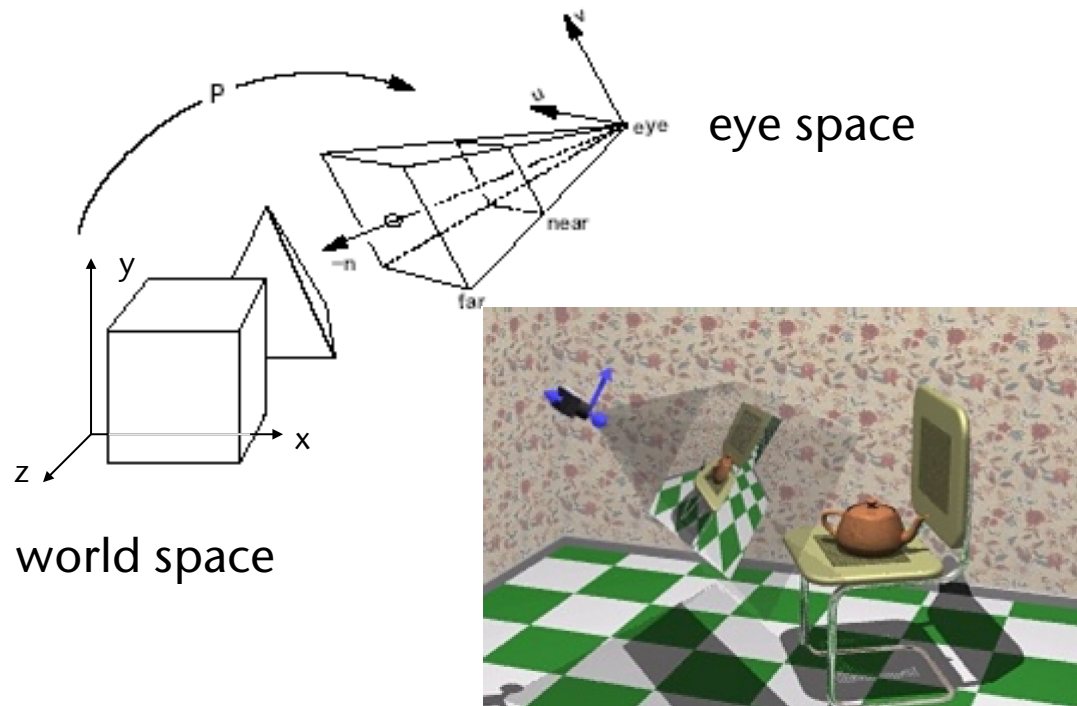
Illumination (Beleuchtung, Schattierung)

- **Beleuchten** von Dreiecken (für Schattierung und Highlights) gemäß der Material-Eigenschaften, Oberflächeneigenschaften und Lichtquellen
- Lokale **Beleuchtungsmodelle** (Diffuse, Ambient, Phong, etc.)



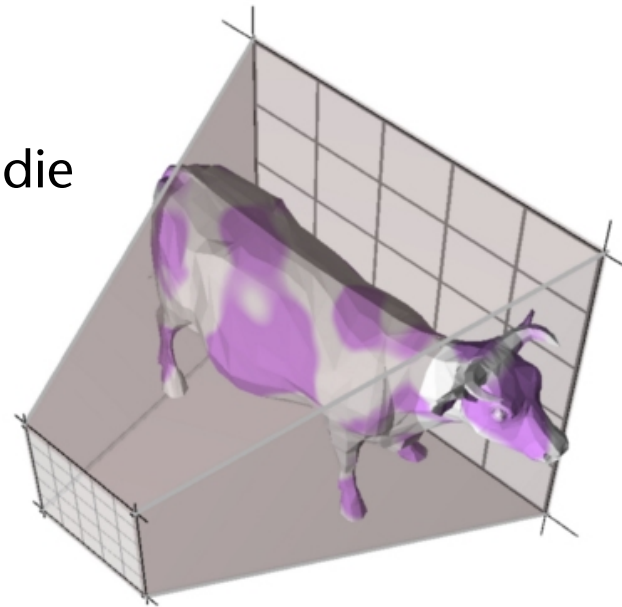
Viewing Transformation

- Umwandeln von **Welt-Koord.** nach **Kamera-Koord.**
- Bestimme Transformation für komplette Szene so, daß Betrachter-Position in den Ursprung verschoben wird und Blickrichtung entlang (-Z)-Achse ist

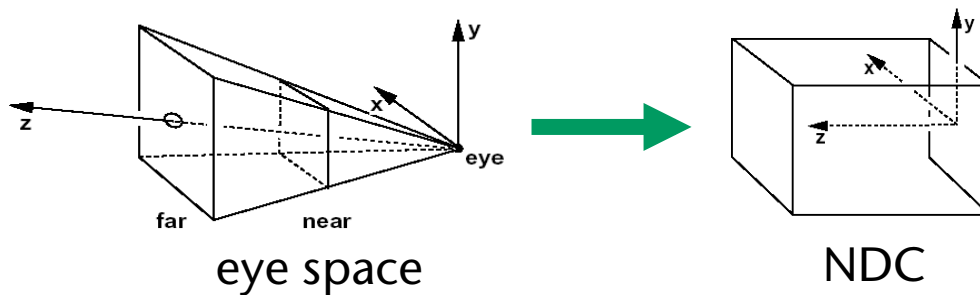


Clipping

- Abschneiden der Polygone, die außerhalb des sichtbaren Bereiches liegen (*view frustum*)



- Transformiere Szene in **normalisierte Koordinaten (NDC)**



Modell Transformation

Illumination (Shading)

Viewing Transformation (Perspective / Orthographic)

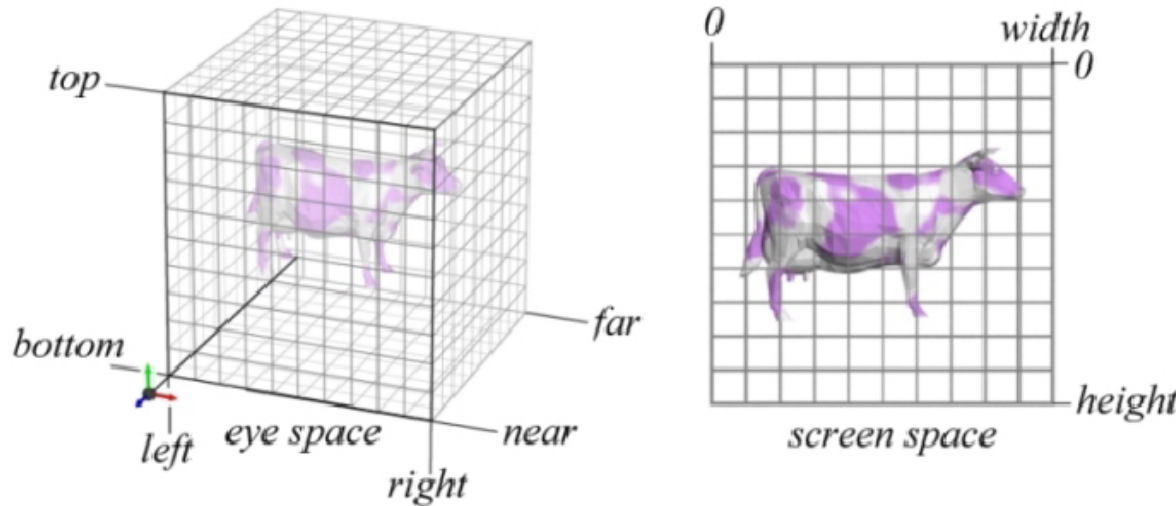
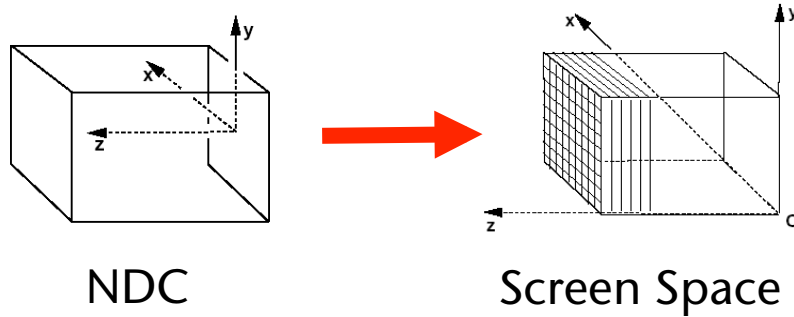
Clipping

Projektion (in Screen Space)

Scan Conversion (Rasterization)

Visibility / Display

- Das Objekt wird in ein 2D Bild umgewandelt (*screen space*) projiziert



Modell Transformation

Illumination (Shading)

Viewing Transformation (Perspective / Orthographic)

Clipping

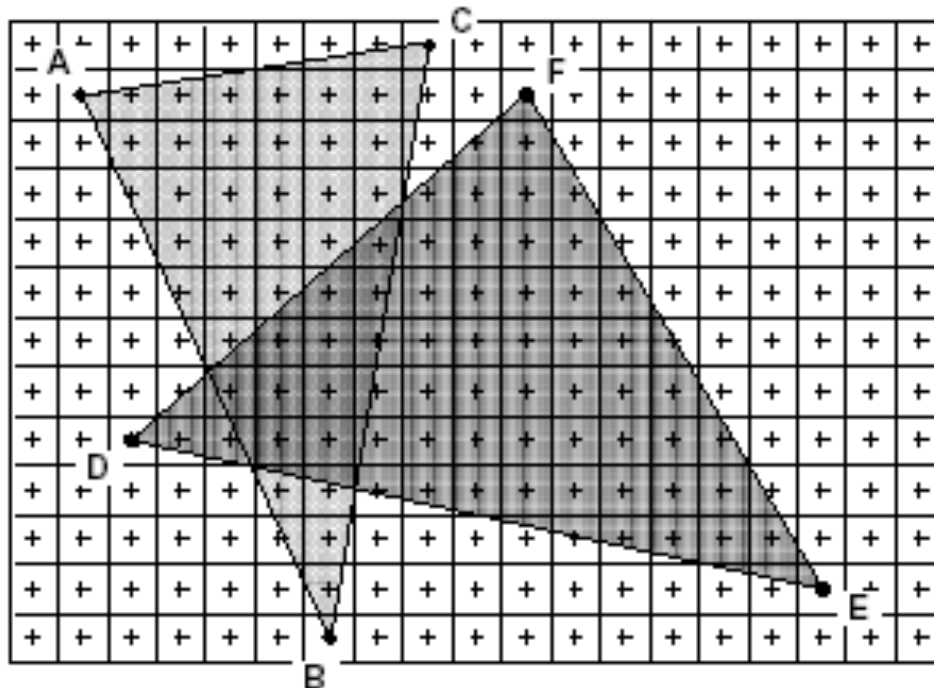
Projection (in Screen Space)

Scan Conversion (Rasterization)

Visibility / Display

Scan Conversion (Rasterisierung)

- Rasterisierung der Polygone in Pixel
- Ecken-Werte interpolieren (Farbe, Tiefenwert, ...)



Modell
Transformation

Illumination
(Shading)

Viewing Transformation
(Perspective / Orthographic)

Clipping

Projektion
(in Screen Space)

Scan Conversion
(Rasterization)

Visibility / Display

- Verdeckungen bestimmen
- Evtl. weitere Pixel-Operationen:
 - Blending mit vorhandenem Frame-Buffer-Inhalt
 - Maskierung (z.B. wegen Verdeckung durch andere Fenster)
 - Farb-Transfer

