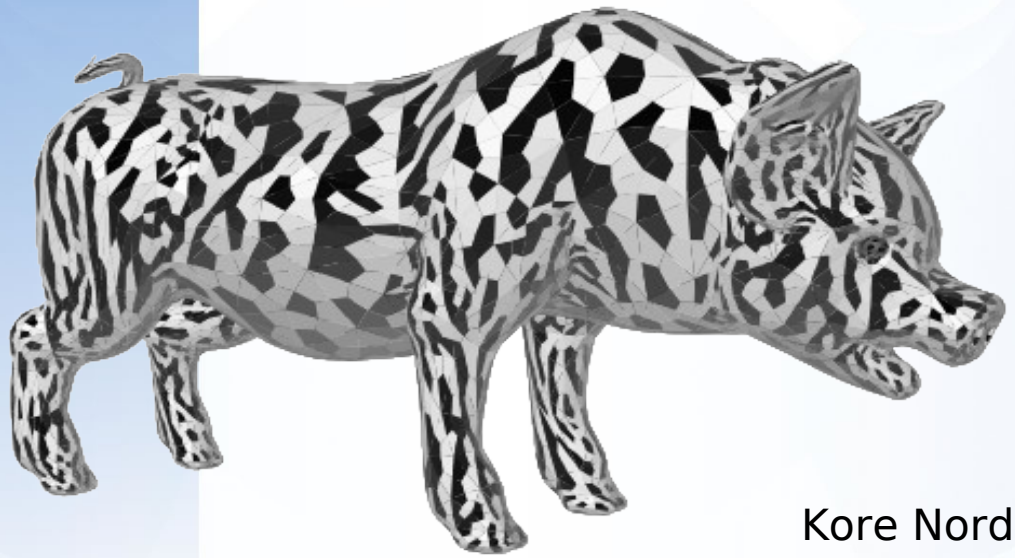
eZ systems

Share your information

3D rendering

Introduction and interesting algorithms

PHP Usergroup Dortmund, Dortmund, 2006-12-14



Kore Nordmann <kore@php.net>

eZ systems

Share your information

Speaker

- Kore Nordmann
  - Studies computer science at the University Dortmund
  - Working as a software developer for eZ systems on eZ components and eZ publish
  - Maintainer and Developer of Image_3D
  - PHPUnit developer

eZ systems

Share your information

Agenda

- Definitionen
- Darstellung von 3D Objekten
- Datenstrukturen für Polygonnetze
- Transformationen
- Abbildung
- Shading
- Subdivision Surfaces
- Ausblick

eZ systems

Share your information

Definitionen

- Bildgenerierung vollzieht sich in drei Schritten
  - Aufbau der Szene
  - Abbildung auf eine 2D-Ebene
  - Ausgabe der 2D-Elemente
- Die Szene ist die lokale Welt, und enthält
  - Kamera
  - Objekte
  - Lichter

eZ systems

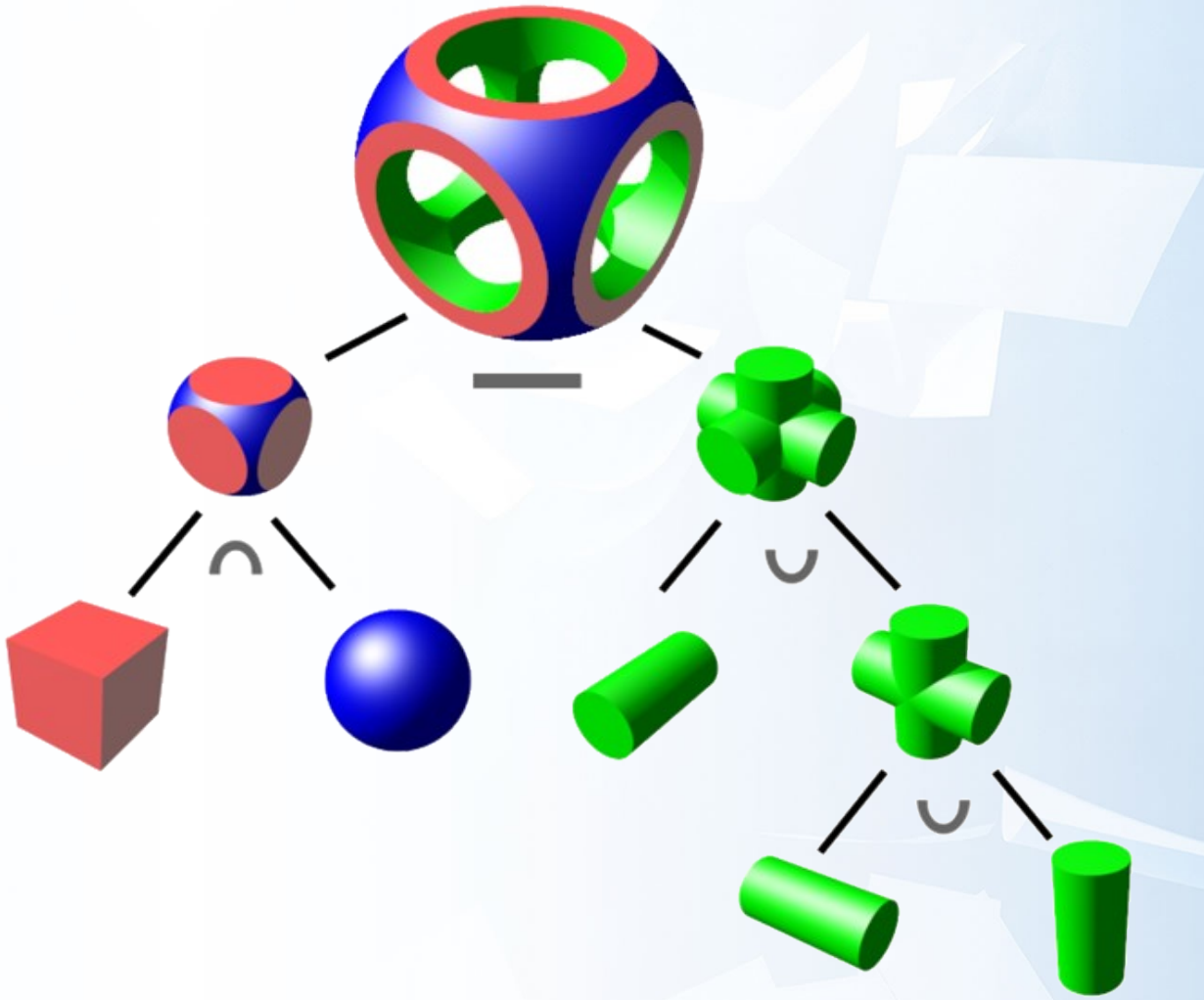
Darstellung von 3D Objekten

Share your information

eZ systems

Konstruktive Körpergeometrie (CSG) (1/2)

- Häufig ind CAD Programmen verwendet
- Oft im Zusammenhang mit Raytracing



Share your information

eZ systems

Konstruktive Körpergeometrie (CSG) (2/2)

- Vorteile
  - Beliebige Detailgenaue Darstellung
  - Geringer Speicherbedarf
- Nachteile
  - Komplexe Algorithmen zur Schnittberechnung
  - Komplexe Abbildung auf 2D-Ebene
  - Rechenintensiv
- “3D-Vektorgrafiken”

Share your information

eZ systems

Share your information

Körper aus Voxeln (1/2)

- Dreidimensionaler Pixel im Raum
- Einzelne Spiele verwendeten Voxel-basierte 3D-Engines
  - Keine Hardware-Beschleunigung
- Verwendung in der verarbeitung von Eingaben aus 3D-Scannern

eZ systems

Share your information

Körper aus Voxeln (2/2)

- Vorteile
  - Triviale Abbildung
- Nachteile
  - Sehr hoher Speicherbedarf
  - Komplizierte Konstruktion
- “3D-Bitmaps”

eZ systems

Share your information

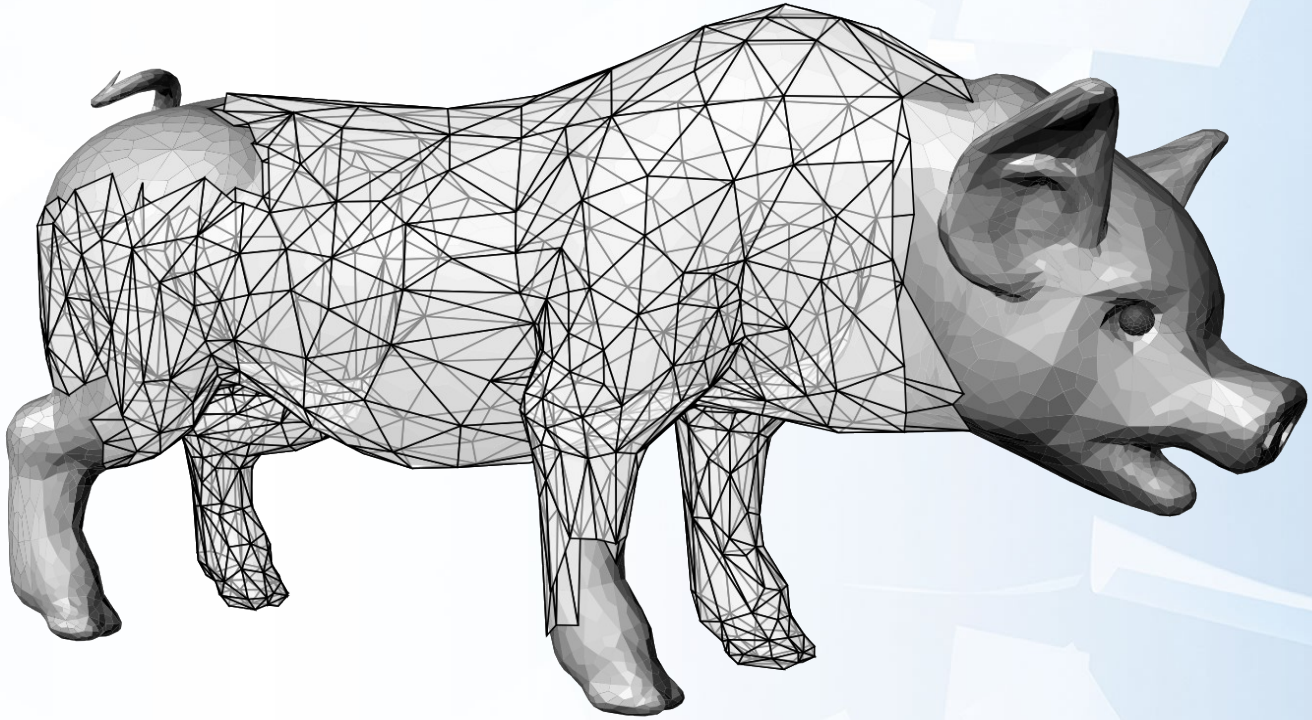
Kompromiss: Polygonnetze (1/4)

- “Ein Polygon bezeichnet ein Vieleck, dessen Punkte durch Kanten miteinander verbunden werden, so dass eine geschlossene Fläche entsteht.”
- Zur algorithmischen Vereinfachung:
  - Beschränkung auf 3 Eckpunkte
  - Reihenfolge der Ecken (im|gegen den) Uhrzeigersinn
- In der 3D-Grafik
  - Eckpunkte liegen immer auf einer Ebene

eZ systems

Kompromiss: Polygonnetze (2/4)

- Zur Interpolation von Objekten
  - Endliche Anzahl von Punkten auf dem Körper verteilen
  - Punkte zu einem Polygonnetz verbinden
  - Je höher die Anzahl der Punkte, desto realistischer das Ergebnis



Share your information

eZ systems

Share your information

Kompromiss: Polygonnetze (3/4)

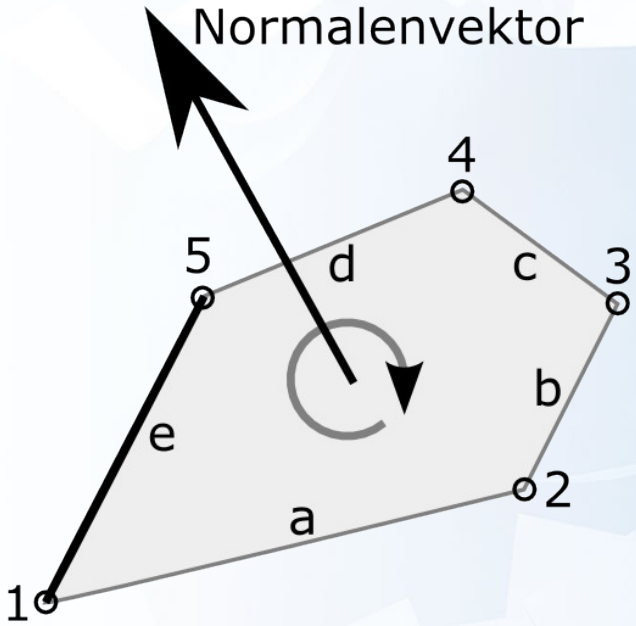
- Üblichstes Implementierungsmodell:
  - Polygone bieten einen guten Kompromiss zwischen Speicherplatzverbrauch und Darstellungsgeschwindigkeit
  - Aufgrund des weiten Gebrauchs in verschiedenen Spielen und Modellierungsprogrammen sind die Algorithmen, die Polygonnetze betreffen hervorragend dokumentiert
  - Es ist bereits eine breite Palette an Polygonmodellen verfügbar

eZ systems

Share your information

Kompromiss: Polygonnetze (4/4)

- Definition über die Kanten oder Ecken mit implizierter Ordnung
  - In Image_3D aggregiert Image_3D_Object eine beliebige Menge von Image_3D_Point
- Reihenfolge der Punkte definiert die Vorderseite / den Normalenvektor eines Polygones



eZ systems

Datenstrukturen für Polygonnetze

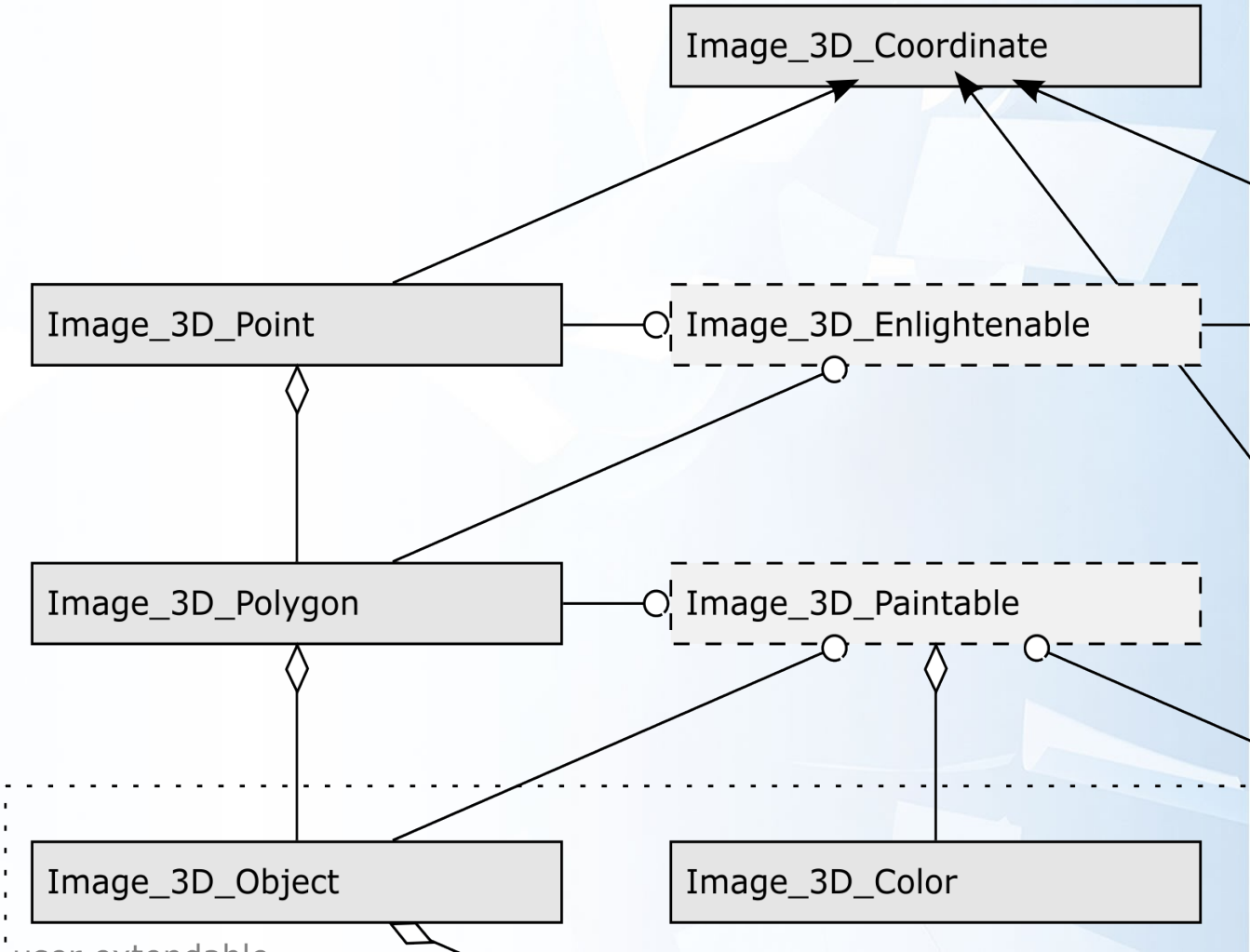
Share your information

eZ systems

Share your information

Naiv: Aggregation von Punkten

- Beispiel Objektmodell in Image_3D
- Code?

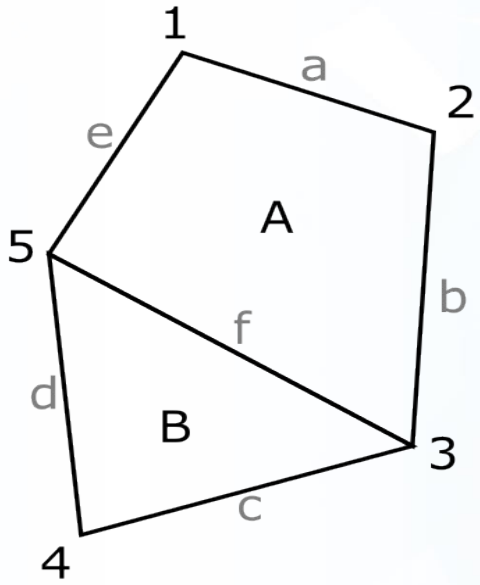


eZ systems

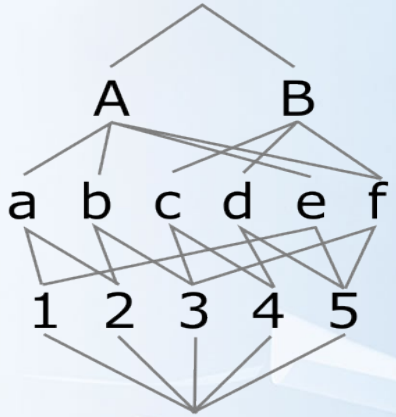
Share your information

Schnell: Knoten-Kanten-Inzidenzgraph

- Speichern der Objekt-Polygon-Beziehungen
- Speichern der Polygon-Kanten-Beziehungen
- Speichern der Kanten-Knoten-Beziehungen
- -> docs/examples/inzidenz.php



Polygone



Inzidenzgraph

eZ systems

Share your information

Transformationen

- Es wird mathematisch...

eZ systems

Share your information

Transformationen

- Neue Objekte liegen in Einheitsgröße in der Szenenmitte
- Anwendbare Transformationen:
  - Drehen
  - Translation (Verschieben)
  - Stauchen
  - Spiegeln
- Diese Transformationen lassen sich in 3x3 Matrizzen ausdrücken

Transformationen von Objekten (1/5)

- Ein Objekt wird transformiert, indem alle seine Eckpunkte transformiert werden
  - In Image_3D reicht die Methode Image_3D_Object::transform() die Transformationsmatrix an seine Polygone, und diese an ihre Punkte durch
- Ein Punkt im dreidimensionalen Raum ist ein Koordinatentripel
  - P = (0, 42, 23.5)
- Jeder Punkt laesst sich auch über seinen Ursprungsvektor darstellen

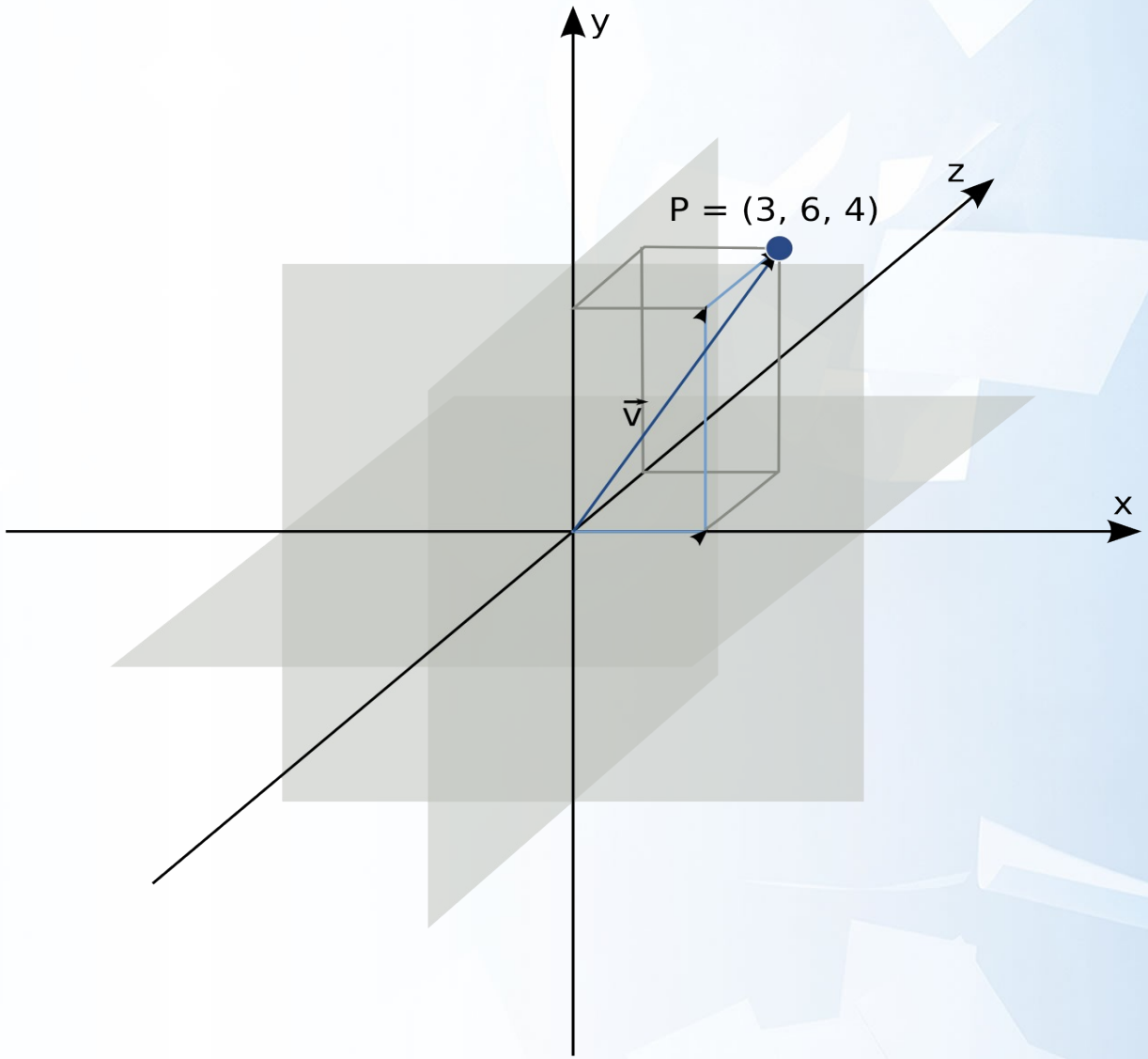
$$v = \begin{pmatrix} 0 \\ 42 \\ 23.5 \end{pmatrix}$$

eZ systems

Share your information

Transformationen von Objekten (2/5)

- Veranschaulichung



eZ systems

Transformationen von Objekten (3/5)

- Transformation
  - Multiplikation des Urpsrungsverktors mit der Transformationsmatrix
- Matrixmultiplikationen machen einen Großteil der Rechenzeit aus
  - Hardwareseitige durchfuehrung in Grafikkarten
- -> docs/examples/matrix.php

eZ systems

Transformationen von Objekten (4/5)

- Verwendung von 4x4 Matrizzen
  - Einheitsmatrix
  - Belegung der ersten drei Spalten / Zeilen
  - Ermöglicht das Multiplizieren mehrerer Transformationsmatrizzen und spätere gemeinsame Anwendung
- -> docs/examples/cone.php

eZ systems

Transformationen von Objekten (5/5)

- Modifikation der Kamera durch Modifikation der ganzen Welt
  - Wenn sich alle Objekte bewegen scheint sich der Betrachter zu bewegen

Share your information

eZ systems

Abbildung

Share your information

eZ systems

Share your information

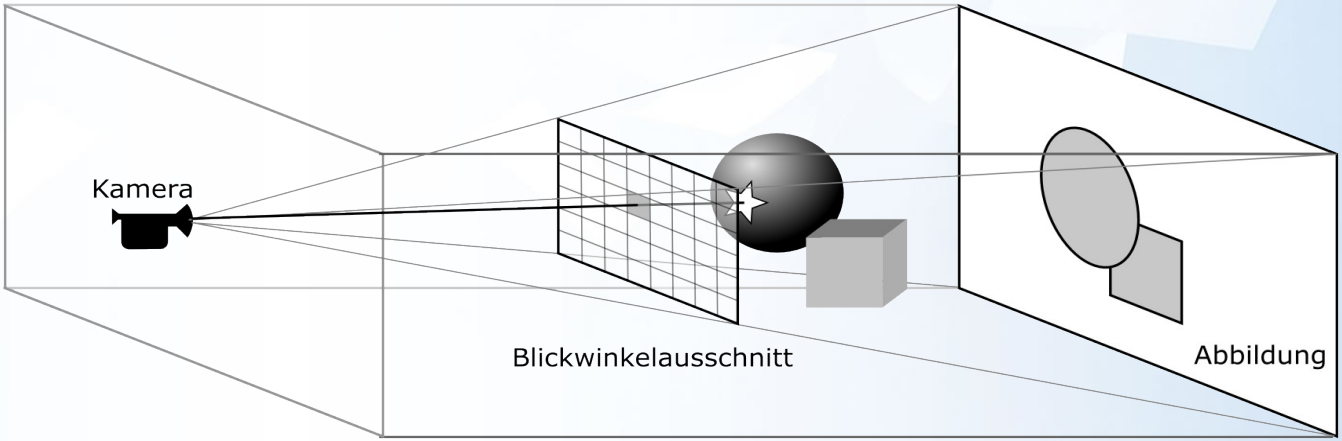
Raytracing (1/4)

- Übersetzung: “Strahlverfolgung”
- Algorithmus basiert auf der Ausbreitung von Strahlen von der Kamera aus
- Relaistische Abbildung durch relativ korrekte Simulation des Lichtweges
  - Licht wird in umgekehrter Richtung verfolgt

eZ systems

Raytracing (2/4)

- Strahlen werden durch ein virtuelles Raster geschickt
- Einfärbung der Pixel aufgrund von Objekt- und Lichtfarbe



Share your information

eZ systems

Share your information

Raytracing (3/4)

- In der Natur geht der Lichtstrahl von der Lichtquelle aus
  - Beim Versenden der Strahlen von der Kamera aus gehen keine Strahlen “verloren”
- Durch rekursive Abarbeitung je nach Spiegelung / Brechung und Transparenz findet eine korrekte Beleuchtungssimulation statt
  - Rekursionstiefe ist gewöhnlich beschränkt

eZ systems

Raytracing (4/4)

- Pseudocode:
  - foreach ($raster as $point)
    - $object = sendRay($point)->getNearestCut();
    - foreach ($lights as $light)
      - if (!sendRay($light)->getNearestCut())
        - » $light->addInfluence($object)
    - if ($object->transparent)
      - raytrace()
    - if ($object->reflect)
      - raytrace()
- -> Image/3D/Renderer/Raytrace.php

Share your information

eZ systems

Share your information

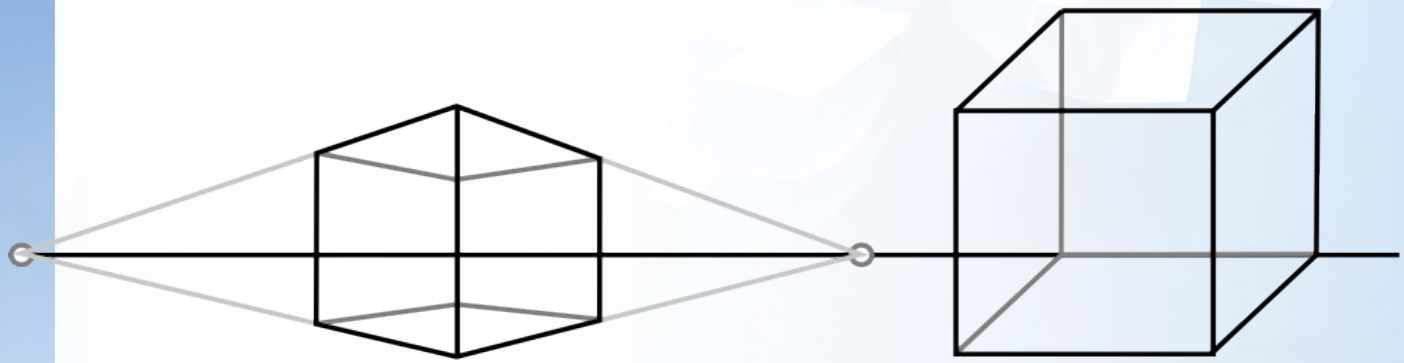
Projektion

- Berechnet für jeden Punkt eines Objektes einen Punkt in der Ebene

eZ systems

Projektionsalgorithmen

- Zentralprojektion entspricht der natürlichen Sicht des Menschen
- Parallelprojektion eignet sich zur Darstellung statistischer Daten



Zentralprojektion

Parallelprojektion

Share your information

eZ systems

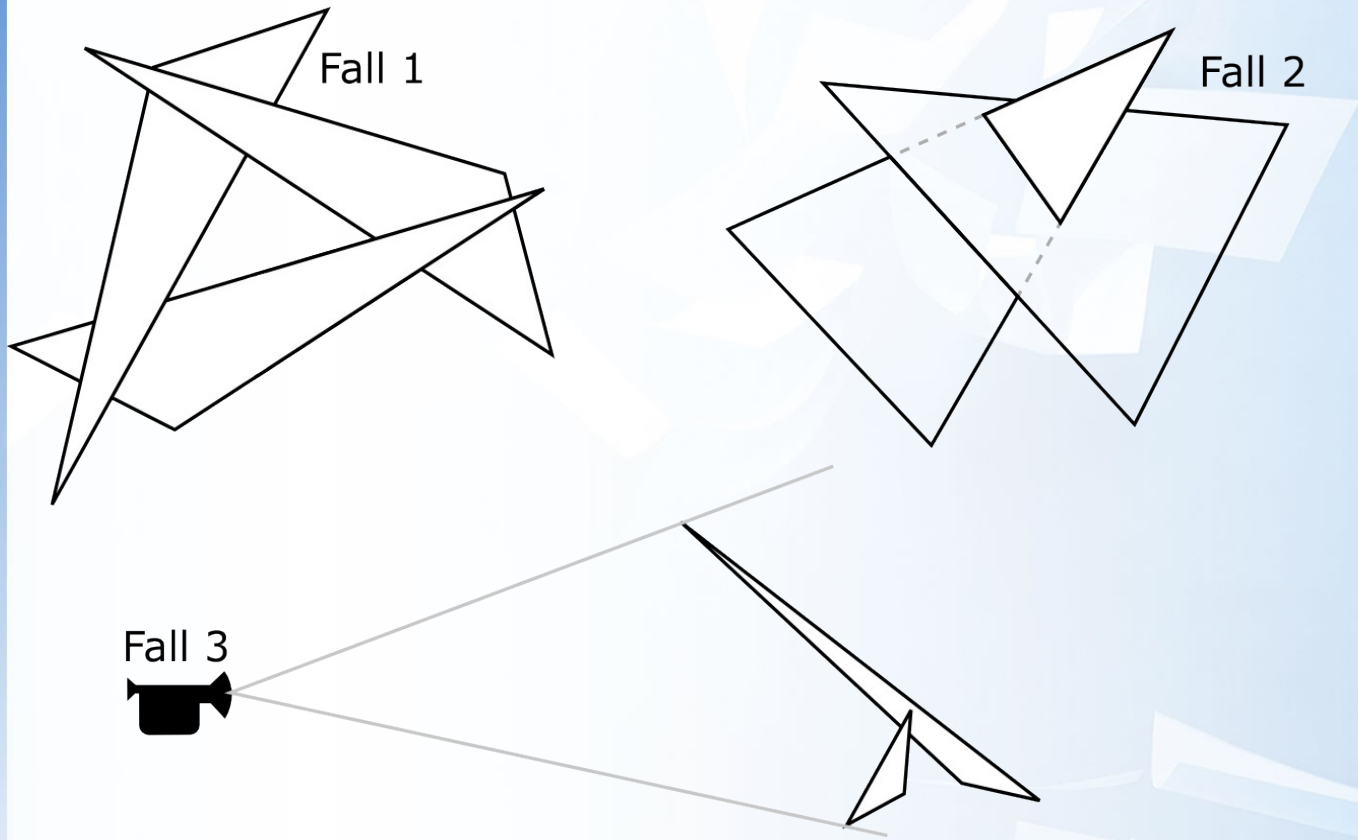
Share your information

Prioritätslistenverfahren

- Nach der Abbildung auf die 2D-Ebene muss definiert werden, in welcher Reihenfolge die Polygone gezeichnet werden
- Entfernung zur Kamera / zum Betrachter
  - Größter / kleinster / mittelerer Z-Wert?

eZ systems

Probleme des Prioritätslistenverfahrens



Share your information

eZ systems

Share your information

Z-Buffering

- Z-Buffer verwaltet eine Prioritätsliste pro Pixel
- Einfachster Fall
  - Zweidimensionaler Array mit Tiefe und Farbe des bislang vordersten Pixels
- Tranzparenzunterstützung
  - Dreidimensionaler Array mit Farbwerten
  - Eventuelle Löschung der Werte hinter deckenden Pixeln (Voxeln ;)
- Nötig fürs Z-Buffering
  - Verrasterung

eZ systems

Share your information

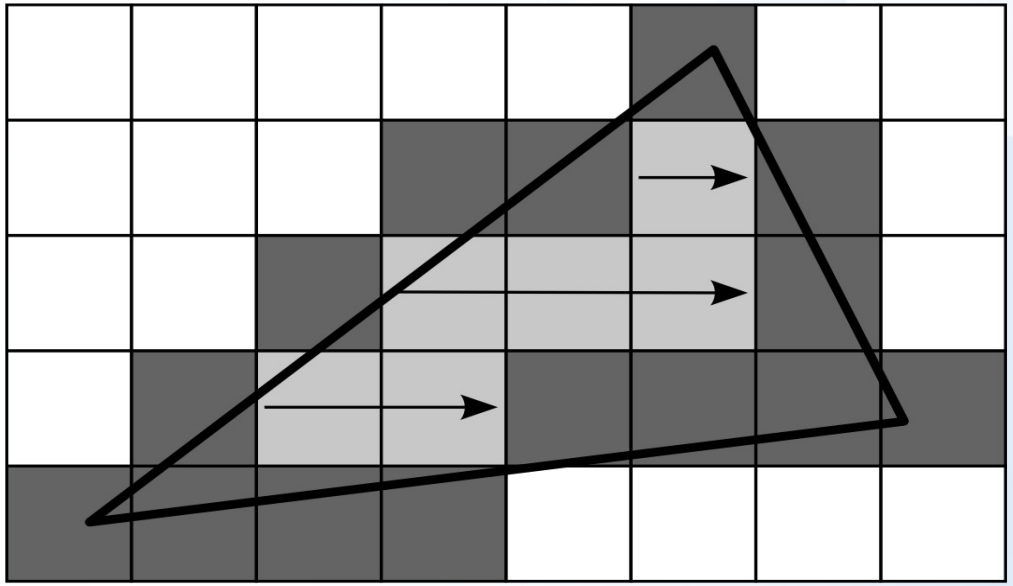
Verrasterung

- “Verrasterung meint die Abbildung eines kontinuierlichen Objektes auf diskrete Werte.”
  - Kontinuierlich: sin-Kurve
  - Diskret: Dirac-Impulse
- Für Polygone ähnlich der trivialen Verrasterung von Linien
  - Schnell: Bresenham-Algorithmus

eZ systems

Verrasterung eines Polygones

- Verrasterung der Aussenlinien
- Auffüllen der umfassten Fläche



Share your information

eZ systems

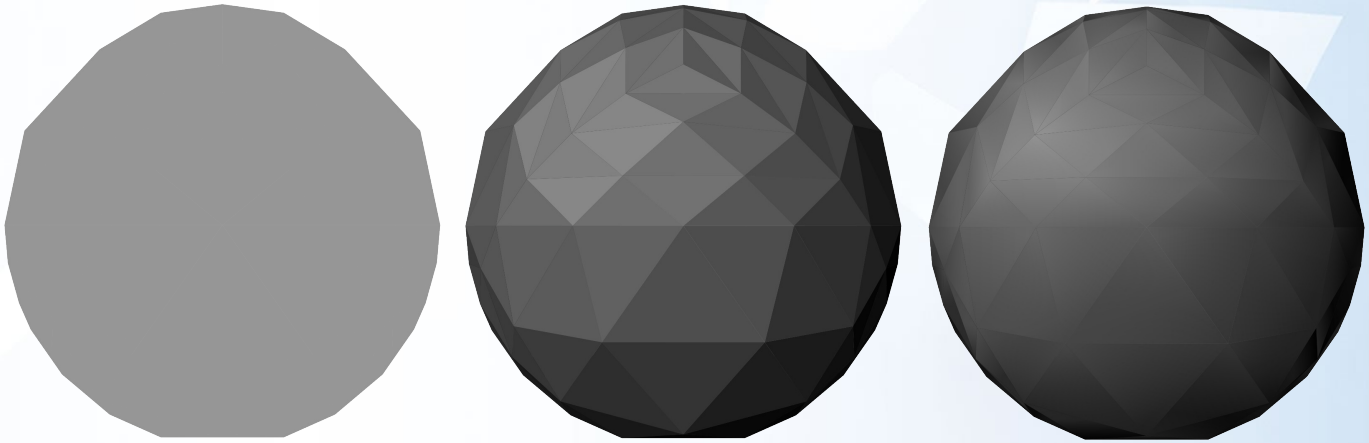
Share your information

Shading

- “Shading bezeichnet das Beleuchtungsmodell, das zur Simulation der Objektoberfläche verwendet wird.”

eZ systems

Shadingtypen in Image_3D



SHADE_NO

SHADE_FLAT

SHADE_GAUROUD

Share your information

eZ systems

Share your information

Shading (1/4)

- No Shading
  - Beleuchtung hat keinen Einfluss
  - Objektfarbe bestimmt alleinig die Ausgabefarbe

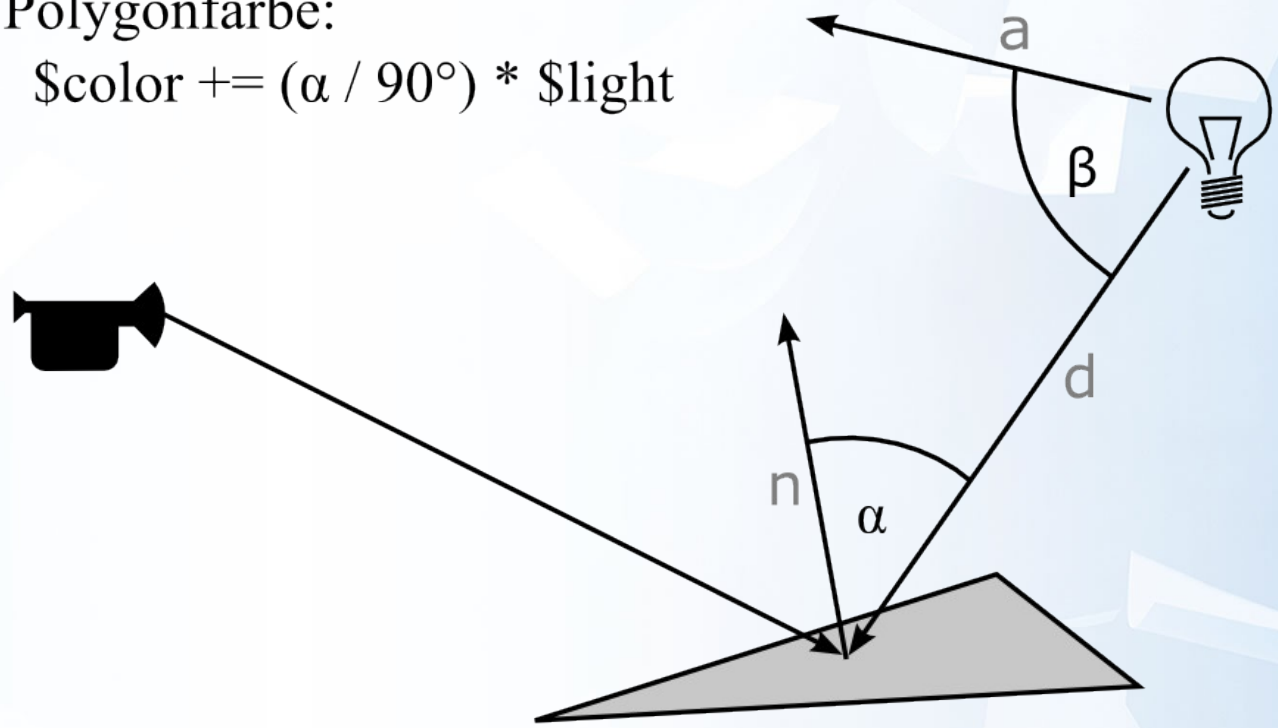
eZ systems

Shading (2/4)

- Flat Shading
  - Zur Beleuchtungsberechnung wird der Normalenvektor des Polygones verwendet

Polygonfarbe:

$color += (α / 90°) * $light



Share your information

eZ systems

Share your information

Shading (3/4)

- Garoud Shading
  - Farbverlauf auf jedem Polygon
  - Verwendung des Normalenvektors eines jeden Punktes
    - Das geht nicht!
  - “Normalenvektor” eines Punktes definiert sich als Durchschnitt der Normalenvektoren der anliegenden Polygone
    - Aneinandergrenzende Polygone müssen dafür die selben Punkte verwenden – keine Kopien
  - SVG-Treiber implementiert in Image_3D als einziger Garoud-Shading

eZ systems

Share your information

Shading (4/4)

- Phong Shading
  - Ermittlung eines “Normalenvektors” für jeden gerasterten Punkt auf einem Polygon
    - Distanzabhängiger Durchschnitt der “Normalenvektoren” der Polygoneckpunkte.
  - Realistische Glanzpunkte
  - Erheblich höhere Anzahl an Beleuchtungsberechnungen
  - Erfordert “manuelle” Verrasterung

eZ systems

Share your information

Subdivision Surfaces (1/2)

- Der Detailgrad von Polygonnetzen ist statisch (maximale Auflösung)
- Bei eintfernten Polygonen ist eine Reduzierung akzeptabel
- Nahaufnahmen sollten eine bessere “Auflösung” bekommen

eZ systems

Share your information

Subdivision Surfaces (2/2)

- Übersetzung: Flächenunterteilung
- “Aufteilung der Polygone in kleinere Einheiten, um den Detailgrad eines Polygonmodelles zu erhöhen”
- Schneller Algorithmus: Catmull-Clark-Algorithmus für Subdiviosn Surfaces

eZ systems

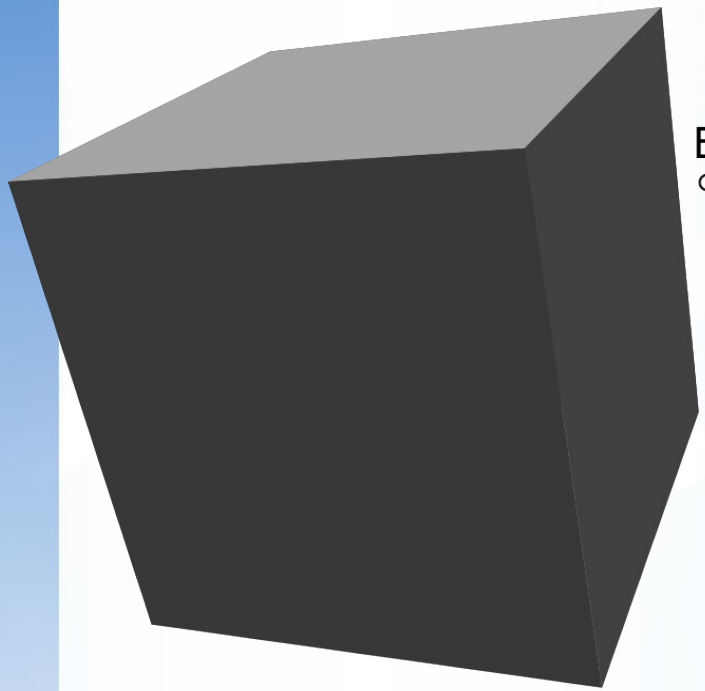
Share your information

Catmull-Clark-Algorithmus (1/4)

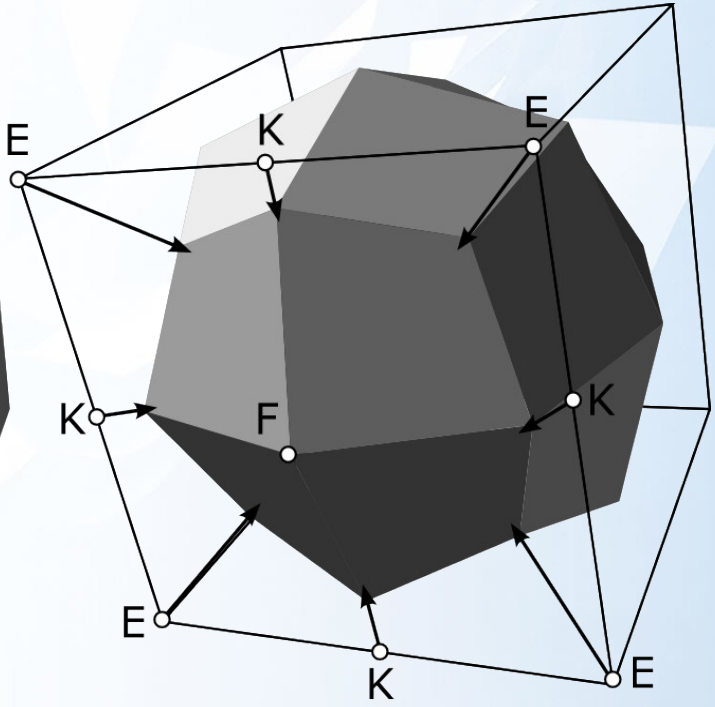
- Operiert auf Knoten-Kanten-Inzidenzgraphen
- Iteration über alle Flächen eines Objektes
  - Erzeugung neuer Punkte
  - Repositionierung existierender Punkte
- Unterscheidung zwischen
  - Flächenpunkte (F)
    - Neuer Punkt in der Flächenmitte
  - Kantenpunkte (K)
    - Neuer Punkt in der Kantenmitte
  - Eckenpunkte (E)
    - Repositionierte existierende Eckpunkte

eZ systems

Catmull-Clark-Algorithmus (2/4)



Original



erste Iteration

Share your information

eZ systems

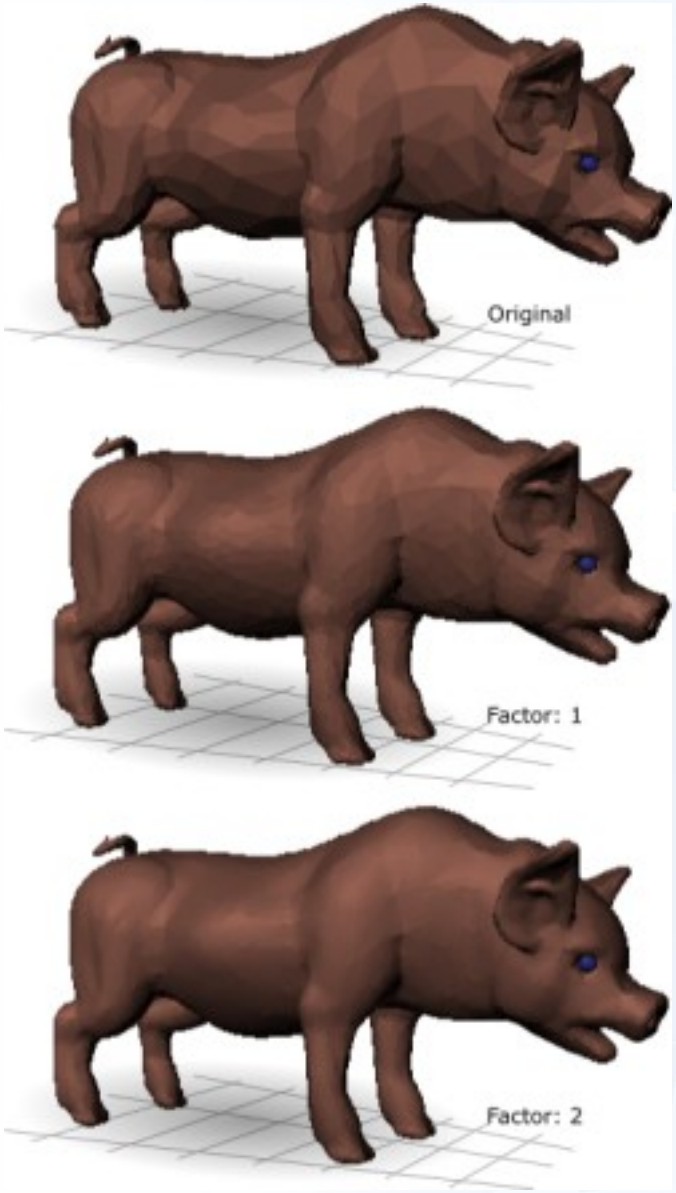
Share your information

Catmull-Clark-Algorithmus (3/4)

- F
  - Durchschnitt der bisherigen Eckpunkte
- K
  - Durchschnitt aus Kantenmitte und F der angrenzenden Flächen
- E
  - Q / n + 2* R / n + S * (n – 3) / n
    - Q = Durchschnitt der angrenzenden F
    - R = Durchschnitt der angrenzenden K
    - S = alter Eckpunkt
    - n = Anzahl der angrenzenden Kanten
- -> Image/3D/Paintable/Object.php

eZ systems

Catmull-Clark-Algorithmus (4/4)



Share your information

eZ systems

Ausblick

Share your information

eZ systems

Share your information

Texturen

- Erfordert manuelle Verrasterung
- Hoher Aufwand beim Clipping und Verzerren von Bitmaps
- Hoher Speicherbedarf der Bitmaps
- Gewöhnlich Hardwareunterstützt und damit problemlos

eZ systems

Share your information

Animationen

- Aneinandereihung von Bildern
- Neu-Erstellung der Szene entfällt
  - Der höchste Aufwand liegt jedoch fast immer beim Rendern
- -> docs/examples/ascii_demo.php

eZ systems

Share your information

Fragen & Fachsimpeln

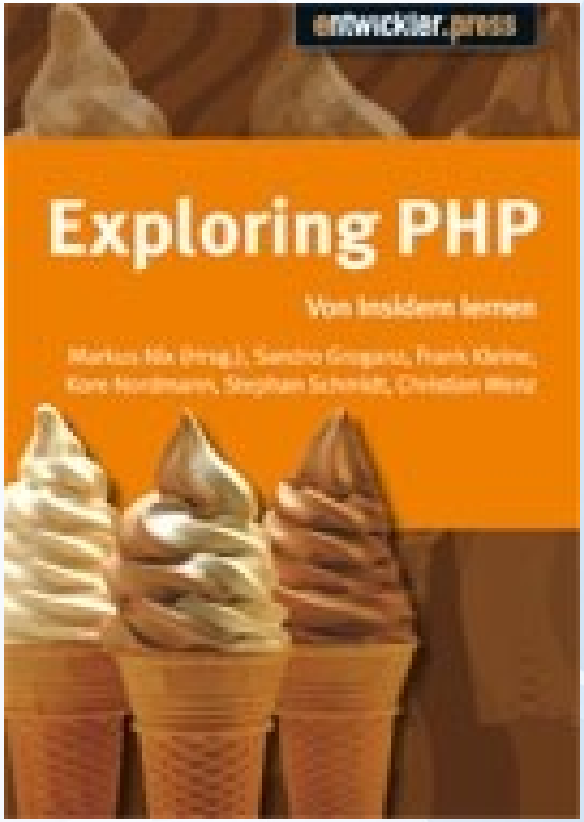
- Interessante Aspekte vernachlässigt?
- Wo sind noch Fragen offen?
- Wann kommt der erste JS-Raytracer?

eZ systems

Share your information

Buchempfehlung

- Exploring PHP
  - Von Insidern lernen
  - ISBN: 3935042957



eZ systems

Share your information

Links

- Ausgezeichnet (deutsche) Artikel auf
  - http://de.wikipedia.org/
- Image_3D:
  - http://pear.php.net/Image_3D
- Homepage
  - http://kore-nordmann.de

Ende