eZ systems

Share your information

3D Rendering mit PHP

- Die neue PEAR-Klasse Image_3D bietet die Möglichkeit nur mit PHP5 3D-Grafiken zu rendern

eZ systems

Share your information

Speaker

- Kore Nordmann
  - Studiert Informatik an der Universität Dortmund
  - Arbeitet als Software Developer für eZ systems
  - PEAR maintainer / developer
- Software Projekte
  - Image_3D
  - eZ components

eZ systems

Share your information

Wo ist Sinn?

- “If it is possible, it will be done.”
- Proof of Concept
- Datenvisualisierung
  - ezcGraph

eZ systems

Share your information

Agenda

- Aufbau der Szene
  - Implementierung in Image_3D
    - Klassenstruktur
    - Interfaces
    - Transformationen
  - Installation
  - Ein eigenes Objekt definieren
  - Beispielszene
- Projektion
- Ausgabe
  - Shading
  - Treiberstruktur
- Performance

eZ systems

Share your information

2.1) Aufbau einer Szene (1)

- Konstruktive Körpergeometrie
  - Mathematische Modelle von Körpern
  - Zusammensetzungen von komplexen Körpern durch Mengenoperationen
  - Verwendung in CAD Anwendungen
- Kritik
  - Beliebig genaue Darstellung
  - Geringer Speicheraufwand
  - Komplizierte Abbildung auf die zweidimensionale Ebene

eZ systems

Share your information

2.1) Aufbau einer Szene (2)

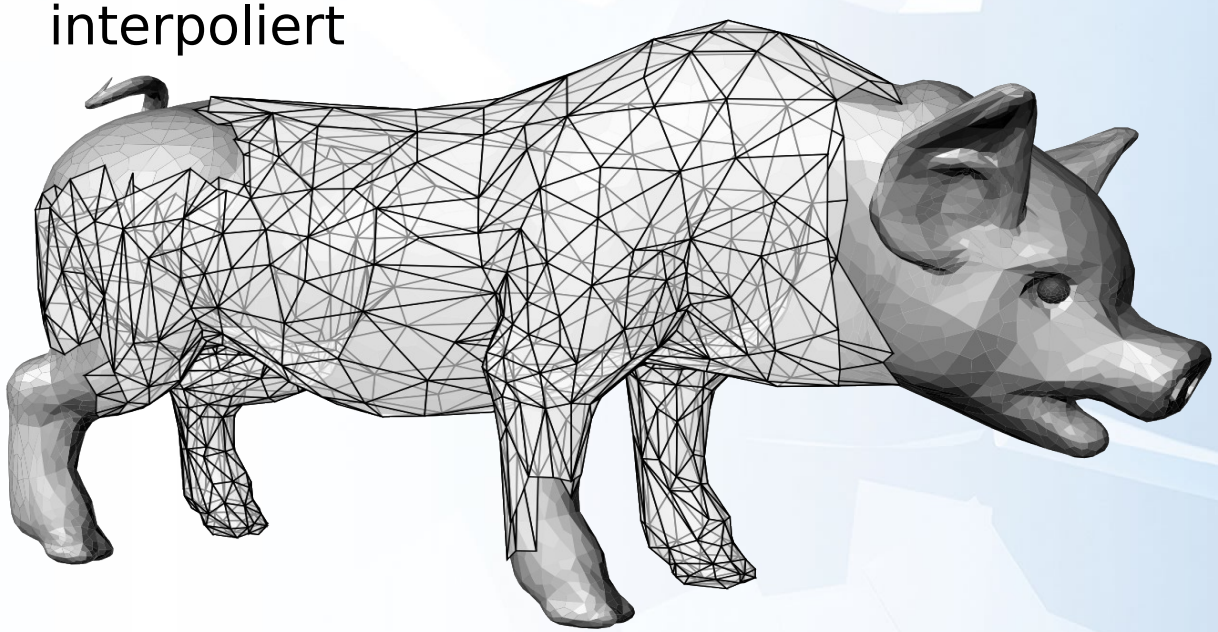
- Voxelgeometrie
  - Dreidimensionaler Pixel im Raum
  - Verteilung von Voxeln lassen sich alle beliebigen Objekte einer Szene darstellen
- Kritik
  - Geringe Auflösung
  - Hoher Speicherbedarf
  - Trotz einzelner Spiele (Outcast) keine Hardwarebeschleunigung

eZ systems

Share your information

2.1) Aufbau einer Szene (3)

- Polygonbasierte Körper
  - Ein Polygon ist eine durch Kanten verbundene Menge an Punkten
  - Menge an Punkten für viele Algorithmen auf drei beschränkt
  - Für die Erzeugung von Körpern sollten die Punkte auf einer Ebene liegen
  - Körper werden durch Polygone interpoliert



eZ systems

Share your information

2.1) Aufbau einer Szene (4)

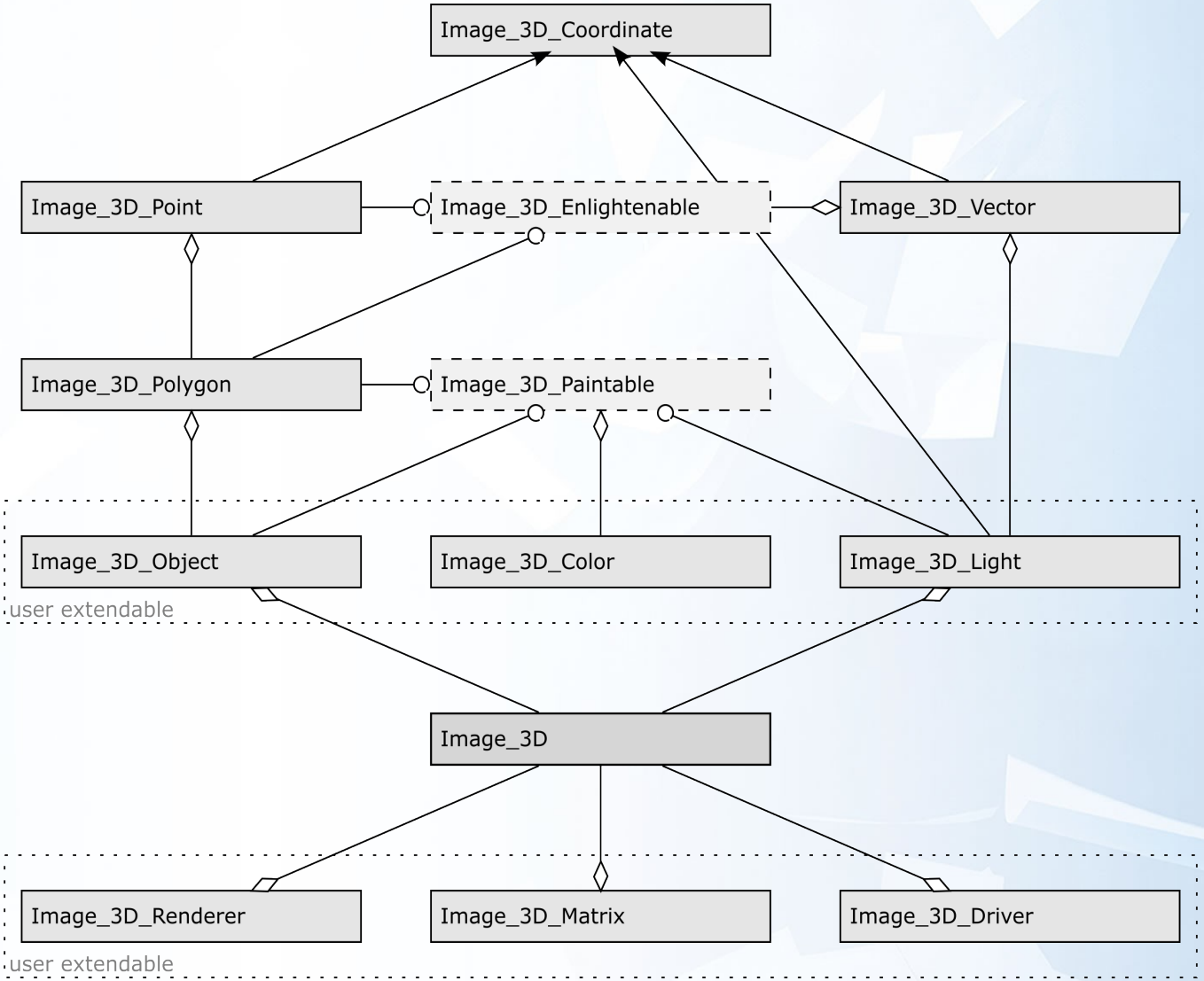
- Kritik
  - Polygone bieten einen guten Kompromiss zwischen Speicherplatzverbrauch und Darstellungsgeschwindigkeit
  - Aufgrund der breiten Verwendung gut dokumentierte Agorithmen
  - Viele Polygonemodelle, auch frei, verfügbar

eZ systems

Share your information

2.2.1) Implementierung

- Klassenstruktur von Image_3D



eZ systems

Share your information

2.2.2) Interfaces

- Image_3D_Enlightenable
  - Implementiert von allen Objekten, die beleuchtet werden koennen
- Image_3D_Paintable
  - Methoden zum Setzen von Farbe oder Material
  - Implementiert von allen Objekten, die

eZ systems

Share your information

2.2.3) Transformationen (1)

- Skalierung, Rotation, Verschiebung, sowie Spiegelung und Stauchung
- Transformationen lassen sich über Transformationsmatrizzen ausdrücken
- Transformation werden auf alle Punkte eines Objektes / einer Szene angewandt

eZ systems

2.2.3) Transformationen (2)

- Factory-Methoden zur Generierung von Transformationsmatrizzen
- Multiplikation mit dem Ursprungsvektor des zu transformierenden Punktes
- `$world->createMatrix('Rotation', array(45, 90, 0));`

$$\begin{pmatrix} 0 & 0 & -1 & 0 \\ 0.707 & 0.707 & 0 & 0 \\ 0.707 & -0.707 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Share your information

eZ systems

Share your information

2.3) Installation

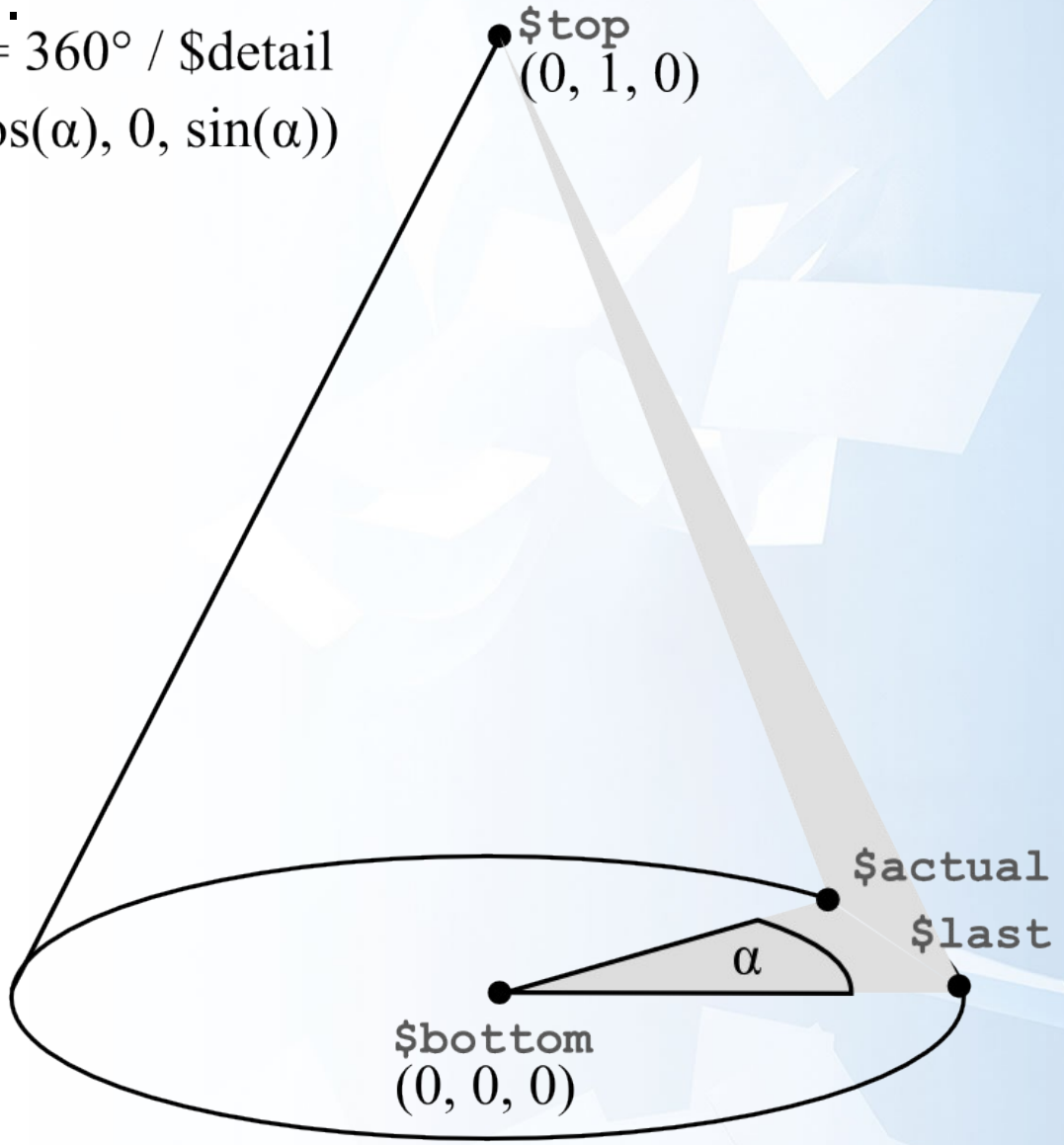
- PEAR ist eine Sammlung von OpenSource PHP Bibliotheken
- Image_3D lässt sich als PEAR Paket über den pear Installer installieren
  - `pear install Image_3D`
- Pakete unter: http://pear.php.net/package/Image_3D

eZ systems

Share your information

2.4) Eigenes Objekt definieren

- Code...



eZ systems

Share your information

2.5) Beispielszene

- Code...

eZ systems

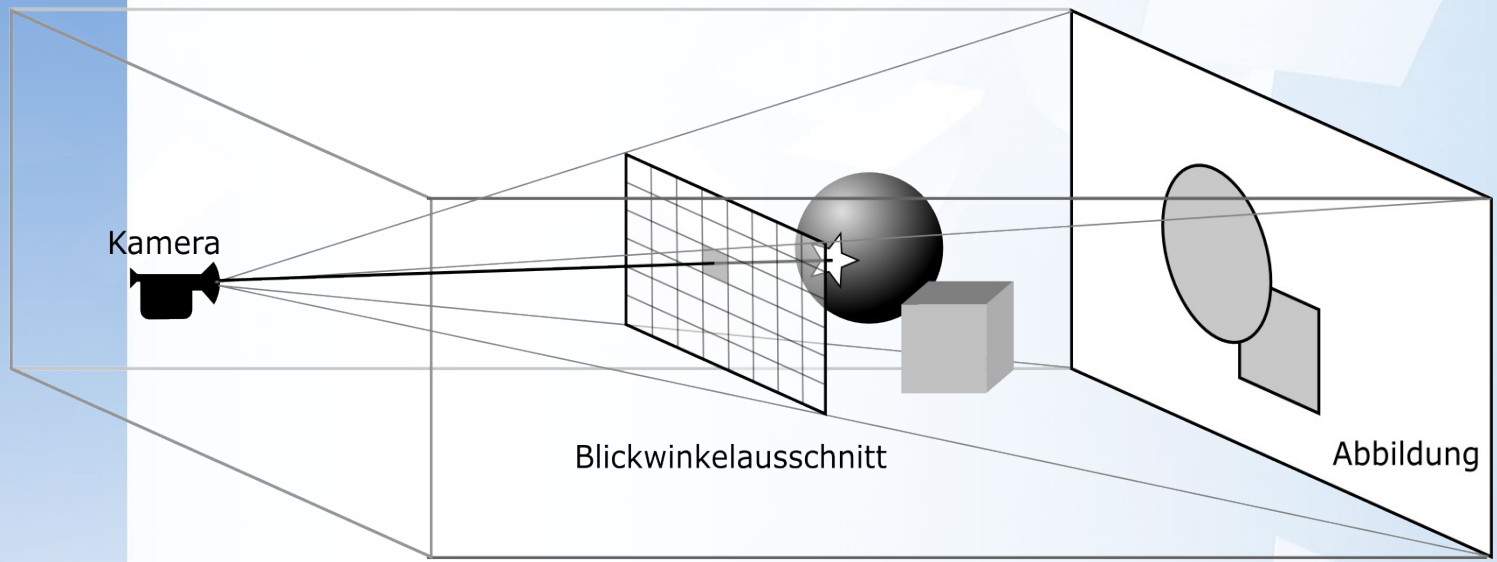
3) Projektion

Share your information

eZ systems

3.1) Raytracing

- Basiert auf Strahlverfolgung

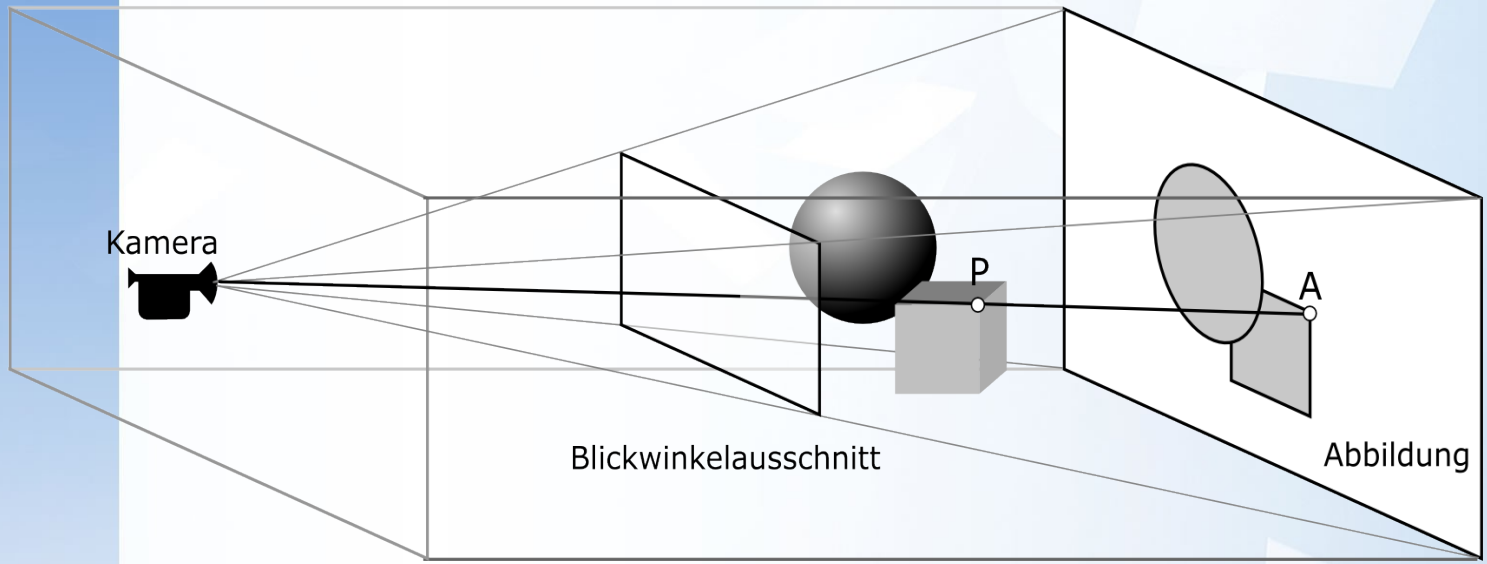


Share your information

eZ systems

3.2) Projektion (1)

- Berechnung der Position auf der Bildebene für jeden Punkt eines Polygones

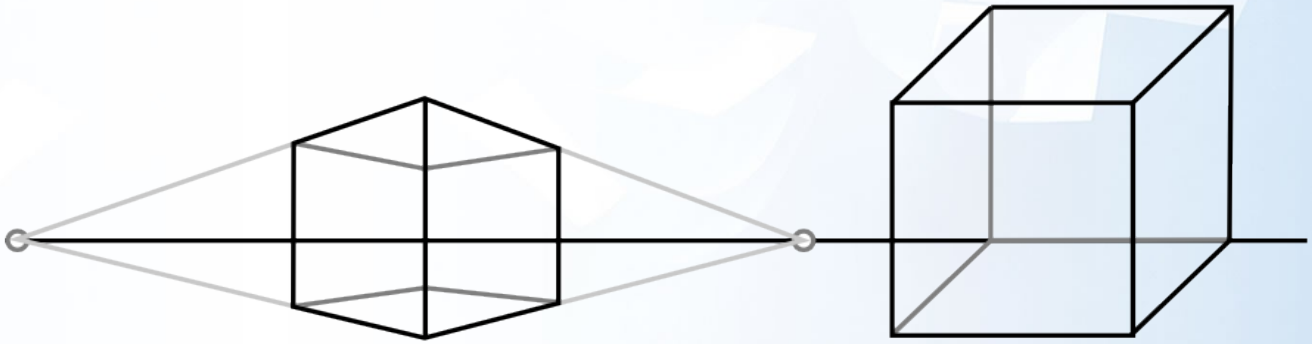


Share your information

eZ systems

3.2) Projektion (2)

- Image_3D_Renderer_Perpectively für zentralprojektion
- Image_3D_Renderer_Isometric für Parallelprojektion



Zentralprojektion

Parallelprojektion

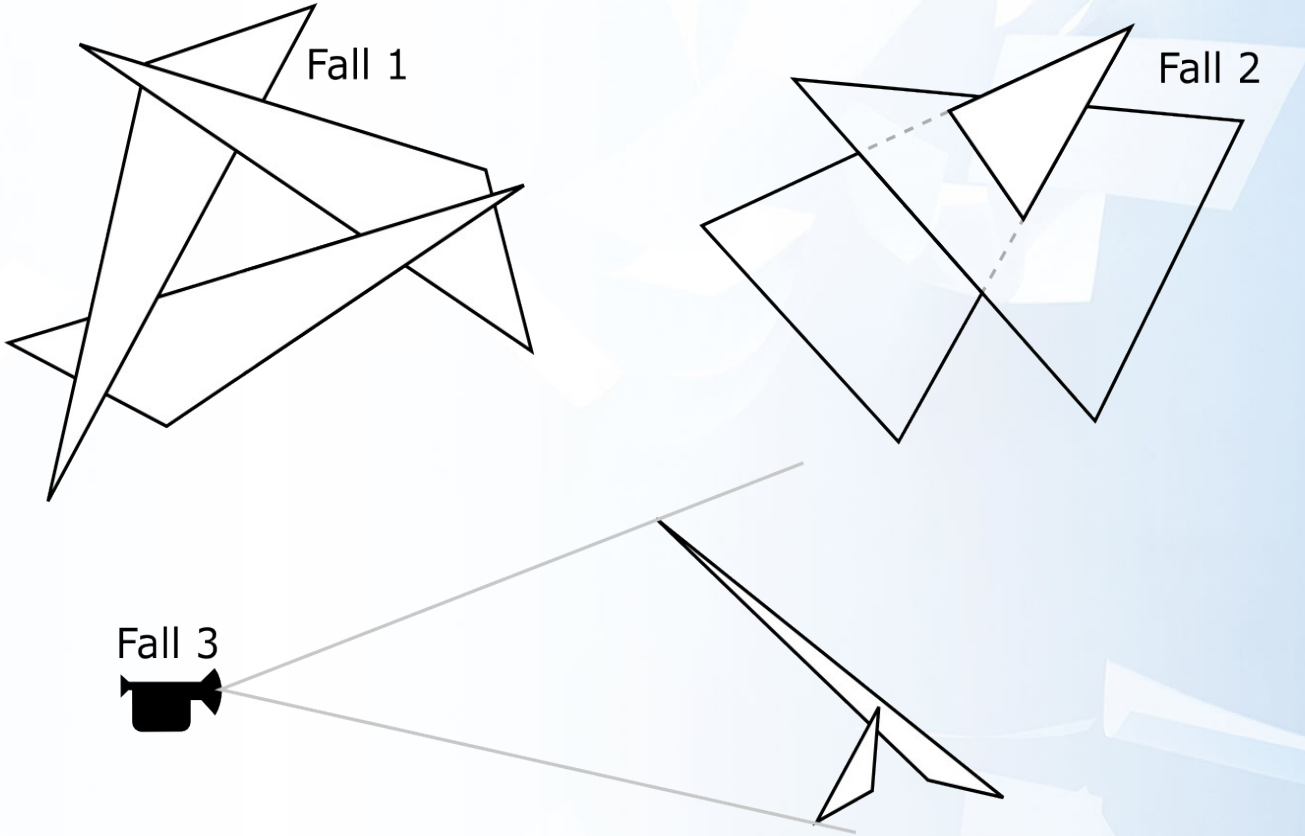
Share your information

eZ systems

Share your information

3.3) Prioritätslistenverfahren

- Bestimmt die Reihenfolge in der die Polygone einer Szene dargestellt werden



eZ systems

Share your information

3.4) Z-Buffering

- Bestimmt die z-Position eines jeden gerasterten Punktes
- Implementiert auf Kosten von Geschwindigkeit in `Image_3D_Driver_ZBuffer`
- Führt zur exakten Darstellung aller Polygonüberschneidungen

eZ systems

4) Ausgabe

Share your information

eZ systems

Share your information

4.1) Ausgabetreiber (1)

- Image_3D stellt Treiber zur Verfügung um verschiendene Ausgabeformate zu generieren
- Treiber unterstützen verschiedene Arten von Shading
- Treiber müssen lediglich Methoden zur Darstellung gefüllter Polygone bereitstellen.

eZ systems

4.1) Ausgabtreiber (2)

- Übersicht

Treiber	*Abhängigkeiten*	*Besonderheiten*
GD	>= GD 2.0	Rendert JPG und PNG Bilder
ZBuffer	>= GD 2.0	Bietet einen Z-Buffer zur exakten Darstellung aller Polygone
SVG	Keine	Rendert SVG Bilder
SVGControl	Keine	Bietet Möglichkeiten zur Transformation der resultierenden SVGs direkt zur Laufzeit
ImageMagick	Imagemagick	Benutzt die externe imagemagick Binary zum Erzeugen des Bildes
ImageCanvas	PEAR::ImageCanvas	Rendert das Bild über die Ausgabeabstraktion von ImageCanvas
ASCII	Keine	Stellt das Bild als farbige ASCII-Grafik auf der Shell dar

eZ systems

4.2) Shading (1)

- Shading bezeichnet das Beleuchtungsmodell, dass zur Simulation der Objektoberfläche benutzt wird

SHADE_NO SHADE_FLAT SHADE_GAUROUD

Share your information

eZ systems

Share your information

4.2) Shading (2)

- No Shading
  - Beleuchtung hat keinen Einfluss auf die Textur einer Oberfläche
- Flat Shading
  - Der Lichteinfluss wird anhand des Normalenvektors eines jeden Polygones berechnet
- Gouraud Shading
  - Lichteinfluss wird anhand eines jeden ein Polygon definierenden Punktes berechnet

eZ systems

Share your information

5) Performance

- Rechenzeit und Speicheraufwand steigen linear mit der Anzahl der Polygone einer Szene

Treiber	*256*	*1024*	*16384*	*65536*
GD	0,11 s	0,32 s	4,30 s	17 s
SVG	0,09 s	0,36 s	5,68 s	35 s
ZBuffer	2,70 s	3,85 s	14,05 s	60 s

eZ systems

Ende

- Links
  - http://pear.php.net/package/Image_3D/
  - http://kore-nordmann.de/

Share your information