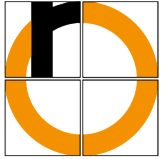




Entwicklung von Computerspielen

Fakultät Informatik
FWPM



3D Modelle

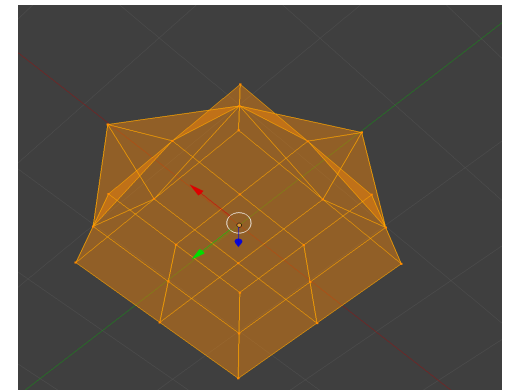
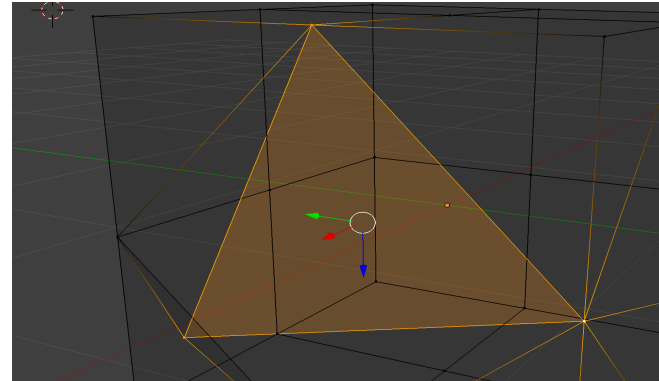
Was ist ein 3D Modell / Asset?

- **Minimum:**
 - Polygon(netz)
 - Textur
- **Zusätzlich:**
 - Zugehöriger Shader
 - Normal Maps
 - Metallic Maps
 - Und mehr

3D Modelle

Was ist ein Polygon?

- Begriff aus der Geometrie:
 - 2 dimensionales Polytop
 - Einfacher: eine Fläche, definiert durch eine Liste von Punkten
 - Beispiele für Polygone: Dreieck, Sechseck, Achteck...
- 3D Modelle bestehen aus einem „Drahtgerüst“, in der Spielentwicklung Polygonnetz oder auch Polygon Mesh genannt. Dieses Mesh (Kurzform) besteht aus mehreren Polygonen, oft aus Dreiecken



3D Modelle

Bestandteile eines Polygonnetzes

- **Knoten (Vertex):**
Definiert die Endpunkte aller Kanten (Punkt muss mindestens 1 Kante haben, sinnvollerweise aber 2)
- **Kanten (Edge):**
Kanten verbinden die einzelnen Punkte und bilden zusammen Flächen.
- **Flächen (Face):**
Einziges in unserem Fall sichtbare Komponente im Spiel, wenn Texturiert.

3D Modelle

Richtwerte an Modelle

Grundlegendes zum Design von Modellen:

➤ **Vertexcount :**

Die Summe aller !sichtbaren! Vertices (Knoten) in einer Szene die Performance drastisch beeinflussen. Als Richtwert:

Bei PCs / Konsolen: nicht mehr als ~ 3.000.000

Mobilgeräten / Low end Hardware: nicht mehr als ~ 200.000

(Keine feste Grenze, schwankt je nach Shader / Renderer, Engine usw. stark. Nur als grober Richtwert zu verstehen)

Character Modelle (sind in aller Regel einige der detailliertesten Modelle) bei AAA Titeln ~ 5.000 - 10.000 Vertices

3D Modelle

Richtwerte an Modelle

- **Durch Backface Culling wird in den meisten Fällen nur die eine Hälfte eines Modells gerendert.**

Nachdem aber Modelle nicht überall die gleiche Vertex Dichte haben (Beispiel Menschlicher Kopf), kann sich dieser Grundsatz stark verschieben.

(Teilweise 95% aller Vertices im Gesicht (Nase, Falten, Mund), 5% für die restliche Struktur (Essentiell nur eine besondere Halbkugel))

- **Mehr dazu in Profiling**

3D Modelle

Ultra Hochauflösende Modelle in Spielen, WIE?

➤ Hintergrund als Textur

Einige Spiele benutzen eine vorgerenderte Skybox, um hochauflösende Hintergründe im Spiel darzustellen und sparen so an Rendering Ressourcen.

➤ Enge Gänge

Eine der „billigsten“ Möglichkeiten den Vertex Count niedrig zu halten, ist es sich in Gängen, die viel verdecken, zu bewegen. Meist sind nur wenige Modelle sichtbar und der Rest kann durch Occlusion Culling weggelassen werden.

3D Modelle

Ultra Hochauflösende Modelle in Spielen, WIE?

➤ Level of Detail Systeme

Eine gerne verwendete Methode zur Darstellung vieler hochauflösender Modelle ist es, mehrere Varianten von einem Modell zu erstellen, jeweils mit geringerer Auflösung.

Weiter entfernte Modelle zeigen dann nur die simpleren Modelle, was bei richtiger Verwendung dem Spieler nicht auffällt.

Unreal Engine: Eingebautes LOD System. Automatisch Erstellung von LOD Stufen.

Unity: LOD System dem für jede LOD Stufe ein anderes / simpleres Modell zugewiesen werden muss.

3D Modelle

Ultra Hochauflösende Modelle in Spielen, WIE?

➤ Tessellation:

Funktioniert durch Anwendung eines Tessellation Shaders.

OpenGL ab Version 4.0 , DirectX ab Version 10

Bei Interesse: <https://www.khronos.org/opengl/wiki/Tessellation>

Grob vereinfacht: einfaches Polygonnetz wird mithilfe des Tessellation Shaders Hochauflösender, da größere Flächen aufgebrochen werden und mit zusätzlichen Dreiecken gefüllt werden.