



# 3D-Grafik-Chips

---

Informatik-Seminar  
Michael R. Albertin  
Betreuer: E. Glatz

## INHALT

<b>1. ZIEL</b> .....	<b>3</b>
<b>2. EINLEITUNG</b> .....	<b>4</b>
2.1. DER UNTERGANG.....	5
<b>3. CHIPGRUNDLAGEN</b> .....	<b>6</b>
3.1. CHIP- & SPEICHERTAKT.....	6
3.2. SPEICHERART .....	6
3.3. RENDERPIPELINE .....	6
3.4. SKALIERBARKEIT.....	7
3.5. SCHNITTSTELLE.....	7
<b>4. FUNKTIONEN</b> .....	<b>10</b>
4.1. RENDERING.....	10
4.2. TRANSFORM & LIGHTING .....	10
4.3. BUMP MAPPING.....	13
4.4. ANTI-ALIASING .....	14
<b>5. SCHLUSS</b> .....	<b>17</b>
5.1. BENCHMARKS.....	17
5.2. EMPFEHLUNG .....	18
5.3. SCHLUSSWORT .....	18
5.4. SCHLUSS AUS ENDE.....	18

## 1. Ziel

Nach dem Studium dieses Dokumentes sollte jeder interessierte Leser die grundlegenden Techniken der 3D-Grafikkarten kennen. Mit diesen Kenntnissen sollte es auch möglich sein, die verschiedenen Chips unterscheiden zu können und Falschdeklaration als solche zu erkennen.

Eine "Falschdeklaration" sehen wir in diesem Beispiel aus dem Weihnachtskatalog 2000 der Firma Microspot:

The image shows a collage of promotional material for a 'GAMER-PC' from a 2000 catalog. It includes:

- A list of features: «Unlimited Fun», Top Multimedia-Prozessor, **Power Game Grafikkarte** (highlighted in red), and Spitzen Soundsystem.
- A detailed component list for the 'Microspot Formula Gamer Pentium III 866': Intel Pentium III 866 MHz Box, 128 MB SDRAM, 30-GB-Harddisk Maxtor DiamondMAX, 8x DVD-ROM-Laufwerk, **Voodoo 4 4500, 32 MB SDRAM** (highlighted in red), Soundkarte 32 Bit Soundblaster PCI, Windows ME, Word 2000.
- A photograph of the PC system with a price tag of 2799.- and 699.-.

In der Kurzbeschreibung zu dem sogenannten Gamer-PC wird von einer Power Game Grafikkarte gesprochen. Betrachtet man jedoch die Komponentenliste, stellt man fest, dass eine Voodoo4 4500-Karte eingebaut wurde. Diese Karte darf heute einfach nicht mehr als Power-Grafikkarte angepriesen werden.

## 2. Einleitung

Im Handel findet man unzählige verschiedene Grafikkarten vieler. Neben den vielen unbekannteren und no-Name Herstellern sind bekannte Firmen wie die folgenden zu finden:

- 3Dfx<sup>1</sup>
- Herkules / Guillemont
- Elsa
- Asus
- Matrox
- ATI
- Creative Labs<sup>2</sup>
- S3<sup>3</sup>
- nVidia<sup>4</sup>



Betrachtet man jedoch die auf den verschiedenen Karten verwendeten Grafikchips, wird man schnell feststellen, dass es nur wenige Chiphersteller auf dem Weltmarkt gibt.

Diesmal ist die Liste nicht ein Auszug, sondern eine fast vollständige Aufstellung:



- 3Dfx
  - Voodoo, Voodoo2, Voodoo Banshee, Voodoo3 2000, Voodoo3 3000, Voodoo4 4500, Voodoo5 5500, Voodoo6 6000<sup>5</sup>
- nVidia
  - Riva128 3D, Vanta, RivaTNT, RivaTNT Ultra, RivaTNT2, RivaTNT2 Ultra, GeForce256, GeForce2, GeForce2 MX, GeForce2 GTS, GeForce2 Ultra, GeForce2 Go, NV2A/NV2X<sup>6</sup>, NV20/NV25<sup>7</sup>
- Matrox
  - Millennium, Mystique, Millennium G450, Millennium G500
- ATI
  - Rage PRO, Rage128, Rage128 PRO, Radeon256
- VideoLogic
  - PowerVR PCX1, PowerVR PCX2, PowerVR 250, Kyro
- S3
  - Virge, Savage 3D, Savage 4, Savage 2000



<sup>1</sup> 3Dfx wurde im Dez. 2000 von nVidia übernommen

<sup>2</sup> Laut Gerüchten möchte Creative Labs aus dem Grafikkarten-Geschäft aussteigen (Stand Jan. 2001)

<sup>3</sup> Keine Aktivitäten auf dem Grafiksektor feststellbar (Stand Jan. 2001)

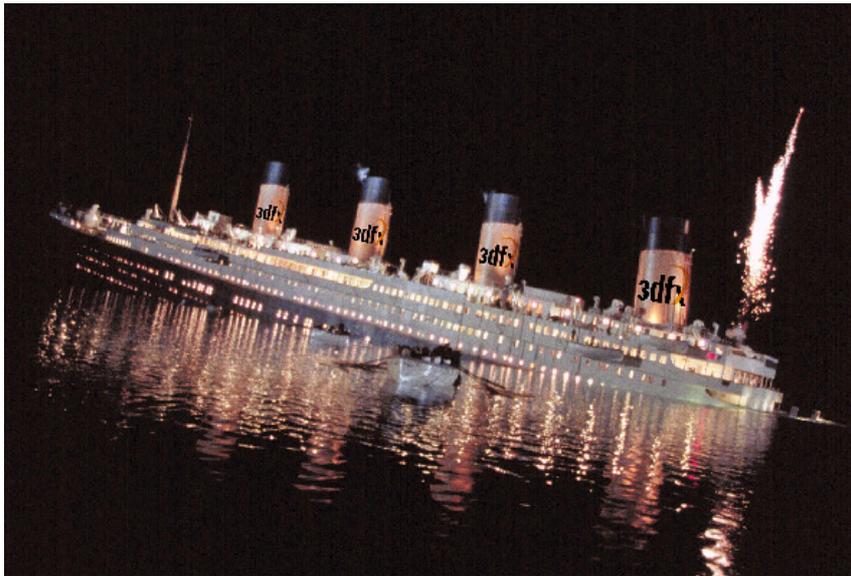
<sup>4</sup> Laut Gerüchten möchte nVidia selbst ins Grafikkarten-Geschäft einsteigen

<sup>5</sup> Die mit Spannung erwartete Spitzenkarte Voodoo 6 6000 wurde kurz vor der Marktreife exklusiv an eine auf Hochleistungsrechner spezialisierte Firma verkauft.

<sup>6</sup> Projektname für Grafikchip, welcher in der Spielkonsole Xbox verwendet werden soll

<sup>7</sup> Projektname für Grafikchips, welche noch 2001 als neue Chipgenerationen auf den Markt kommen sollten. Möglicherweise sind diese Chips schon mit 3Dfx-Technologien ausgestattet.

## 2.1. Der Untergang



Mit diesem Bild wurde das Ende des traditionsreichen und vielerorts beliebten Grafikkarten- sowie Chip-Herstellers 3Dfx kommentiert. Obwohl die Firma mit den Karten Voodoo4-6 etwas über ein Jahr im Rückstand war, glaubten alle an ein Comeback. Selbst zwei Wochen vor der Übernahme durch nVidia wurden noch Prototypen von neuen Karten an Testmagazine verteilt. Selbst Kartentreiber mit Hardwareunterstützung für folgende Generationen tauchten in Fachkreisen auf. Darum schlug die Meldung vom 16.12.00 wie eine Bombe ein:  
nVidia kauft 3Dfx!

Dazu die Pressemitteilung von nVidia (16.12.00):

NVIDIA und 3dfx haben heute ein Abkommen geschlossen, in dem vereinbart wurde, dass NVIDIA bestimmte Grafik-Vermögenswerte von 3dfx, einem Pionier und anerkannten führenden Unternehmen im Bereich von Grafiktechnologie, kauft. Zu diesen Vermögenswerten gehören, aber sind nicht beschränkt auf, alle Patente, alle angemeldeten aber noch ausstehenden Patente, Warenzeichen, Markennamen und der Chipbestand, der zum Grafikprozessorsektor von 3dfx gehört.

Wie aus der Mitteilung herausgeht, wird nVidia weder die Produktionsanlagen, noch die Belegschaft übernehmen. Bis auf wenige Support-Angestellte wird 3Dfx die gesamte Belegschaft entlassen und sich aus dem Grafikkarten-Geschäft zurückziehen.

### 3. Chipgrundlagen

#### 3.1. Chip- & Speichertakt

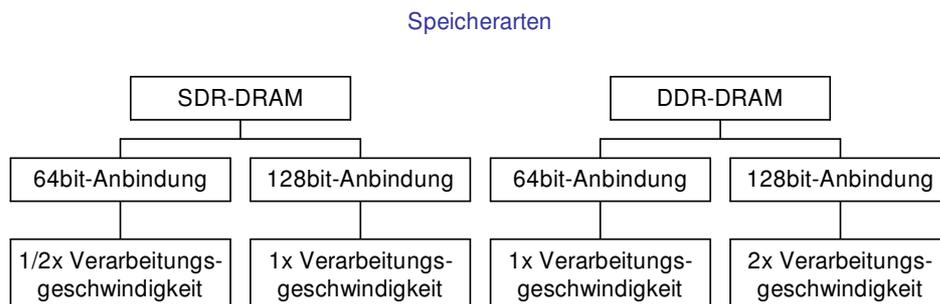
Ein wichtiger Punkt bei der Leistung einer Karte ist die Verarbeitungsgeschwindigkeit des Chips. Wie aus dem CS-Unterricht bekannt, ermöglicht ein hoher Chiptakt eine schnelle Verarbeitung der Daten. Damit der Chip seine Daten auch genügend schnell bekommt und wieder los werden kann, muss natürlich der Speichertakt mindestens gleich schnell sein wie der des Chips.

#### 3.2. Speicherart

Momentan werden zwei Arten von Grafikspeichern auf den Karten verwendet. Die eine Art ist uns vom Arbeitsspeicher her bekannt und nennt sich SDR-DRAM. Diese Bausteine übertragen pro Taktzyklus ein Datenpaket. Gleich zwei Datenpakete pro Taktzyklus übertragen die DDR-DRAM-Bausteine. Sie besitzen damit eine doppelte Übertragungsrate und sind teuer.

Leider wird jedoch mit DDR gespielt. Das heisst, gewisse Hersteller bauen Karten mit DDR-DRAM, binden das RAM aber nur mit halber Busbreite auf die Karte an. So wird die theoretisch doppelte Leistung durch die halbe Busbreite wieder beschnitten.

Die folgende Grafik zeigt die möglichen Kombinationen von Anbindung und Speicherart:



Um eine aussagekräftigere Einheit zu erhalten, wurde der Begriff Bandbreite eingeführt. Die Bandbreite berechnet sich so: Speichertakt \* Speicheranbindung \* Anz. Datenpakete

#### 3.3. Renderpipeline

Neben der Taktung ist für die Geschwindigkeit eines Chips auch die Pipeline-Struktur von entscheidender Bedeutung. Moderne Chips besitzen 2-4 parallel arbeitende Pipelines mit jeweils max. 3 Textur-Stufen.

Ein Radeon-Chip kann z.B. mit seiner 2x3-Architektur in einem Takt zwei Pixel mit je 3 Texturen versehen, die Geforce2 mit 4x2-Architektur 4 Pixel, jedoch mit nur 2 Texturen berechnen.

		Texture Units Per Rendering Pipeline		
Textures Per Pixel		One	Two	Three
→	1 Bilinear	1 pass	1 pass	1 pass
	1 Trilinear	1-2 pass es	1 pass	1 pass
	2 Bilinear	2 pass es	1 pass	1 pass
→	1 Bilinear + 1 Trilinear	2-3 pass es	1-2 pass es	1 pass
→	3 Bilinear	3 pass es	2 pass es	1 pass

GeForce2  
 Radeon

**Radeon: 2 Pipelines à 3 Texturstufen (Units)**  
**GeForce2: 4 Pipelines à 2 Texturstufen (Units)**

Die Grafik zeigt ein Vergleich der 1-3stufigen Textureinheiten pro Pipeline. Sollte eine 3D-Szene mit 3 Texturen pro Pixel gerendert werden, muss die GeForce2 einen zusätzlichen Durchgang aufwenden, um das gleiche Resultat zu erhalten. Dafür kann die GeForce2 gleichzeitig die doppelte Anzahl Pixel berechnen. Welche Architektur also besser ist, kann nicht einfach so bestimmt werden.

Wieso jedoch mehrere Texturen verwendet werden, möchte ich später bei den Grafikfunktionen erklären.

### 3.4. Skalierbarkeit

Reicht die Leistung eines Chips nicht aus, müssen mehrere die benötigte Power liefern. Früher, in der Voodoo2-Zeit, konnte die Leistung nahezu verdoppelt werden, indem eine weitere Voodoo-Karte ins System installiert wurde. In dieser Kombination rechnete eine Karte die geraden Zeilen, die andere die ungeraden Zeilen. Durch das Durchschleifen des Monitorsignals durch zwei Zusatzkarten wurde das Bild aber unsauber.

Dies war zudem teuer, da die zweite Karte genau soviel kostete wie die erste, und heute ist diese Variante mit nur einem AGP-Port gar nicht mehr möglich.

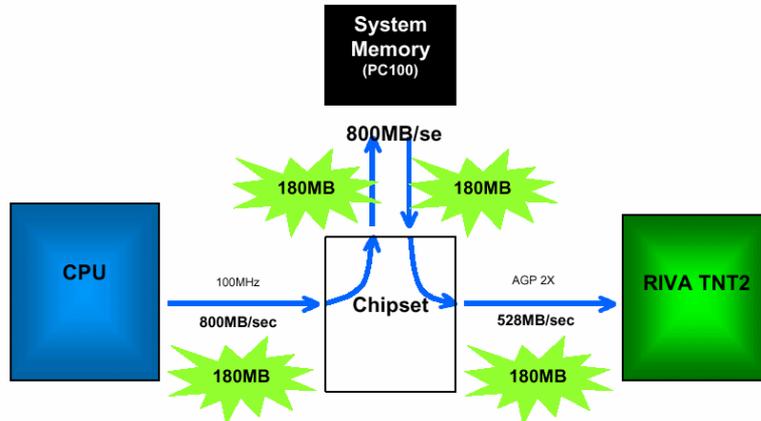
Um den leistungsschwachen Voodoo4-Chip aufzumöbeln, wurden für die Voodoo5 einfach zwei solcher Chips verbaut. Die nie im Handel erschienene Voodoo6 ist mit 4 Chips bestückt. Der Nachteil dieser Methode ist, dass nur der Hersteller die Skalierung vornehmen kann und vorhandene Karten nicht nachgerüstet werden können.

### 3.5. Schnittstelle

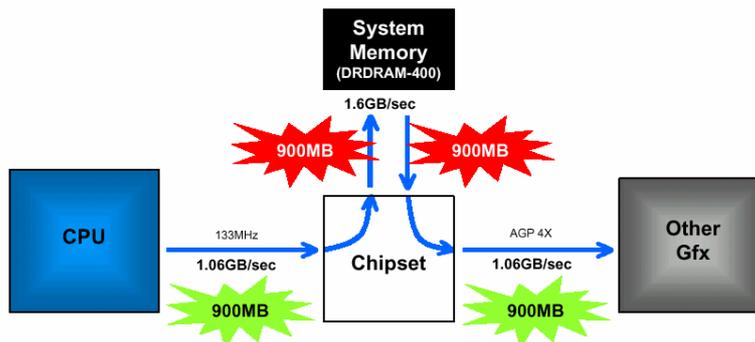
Am Anfang der Geschichte kannte man nur den ISA-Bus, danach auch noch den EISA. Doch dieser war schnell zu langsam und es wurde der schnellere PCI-Bus entwickelt. Karten wie TNT, Voodoo oder Mystique arbeiteten mit dieser Schnittstelle.

Moderne Grafikchip sind heute so leistungsoptimiert, dass auch diese Schnittstelle zu langsam wurde. Speziell für Grafikkarten wurde also der AGP-Bus eingeführt, anfangs mit einfacher oder zweifacher, später mit 4-facher Geschwindigkeit.

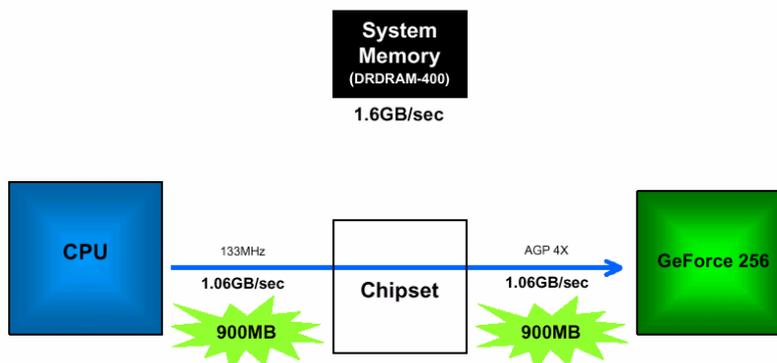
Die neuesten nVidia-Chips sind jedoch so leistungsfähig, dass nicht einmal AGP 4x reicht um die geforderten Datenmengen zu transportieren. Deshalb hat nVidia das Fast Writes-Verfahren eingeführt, welches den Arbeitsspeicher umgeht. Die folgenden Grafiken zeigen schematisch die verschiedenen Verfahren.



Diese Abbildung zeigt ein AGP 2x-System mit einer TNT2-Grafikkarte. Diese Karte berechnet pro Sekunde 2 Mio. Dreiecke, welche je 90 Bytes Daten umfassen. Dies ergibt eine Datenrate von 180 MB/s.



Mit den neuen Chipgenerationen ist die Rechenleistung stark gestiegen. So kann heute eine Rechenleistung von 10 Mio. Dreiecken geboten werden. Dies ergibt bei gleichbleibenden 90 Bytes pro Dreieck eine Datenrate von 900MB/s. Da die Daten jedoch nicht nur durch den AGP-Bus müssen, sondern auch noch über das System-Memory gehen, haben wir ein Nadelöhr.



Mit dem Fast Writes-Verfahren wird das System-Memory einfach umgangen. So lässt sich die Datenmenge von 900MB bewältigen.

Zukünftige Chipgeneration müssten hier ganz klar ansetzen. Eine weitere Leistungssteigerung ohne Reduktion der Datenmenge ist nicht mehr möglich. Dazu gibt es bereits verschiedene Ansätze, wovon eine davon schon jetzt im Kyro-Chip eingesetzt wird.

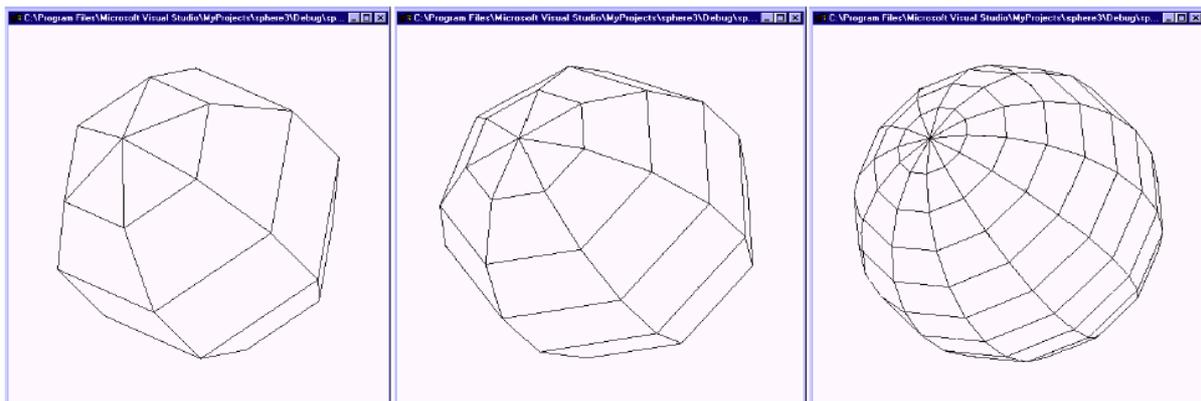
## 4. Funktionen

### 4.1. Rendering

Seit Anbeginn der 3D-Grafik muss man sich zwischen Geschwindigkeit und Darstellungsqualität entscheiden. Die von allen Karten unterstützte 16bit-Farbtiefe ist zwar sehr schnell, in manchen Spielprogrammen wirken die Farbeffekte jedoch nicht optimal. Deshalb gibt es auch noch die 32bit-Farbtiefe, welche speziell in Farbeffekten überzeugen kann, jedoch die Frameraten um bis zu einem Drittel sinken lässt.

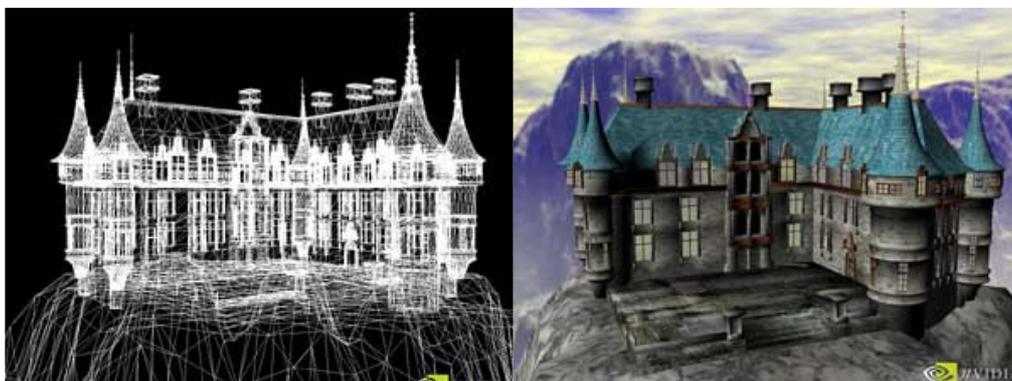
### 4.2. Transform & Lighting

Transform und Lighting sind die ersten beiden der vier grossen Taktschritte, die ein Grafikchip in einer Rederpipeline verarbeitet. Sie sind sehr verarbeitungsintensiv, d. h. sie sind mit einem Set sehr spezifischer mathematischer Anweisungen ausgestattet, die Milliarden mal pro Sekunde agieren, um eine Szene zu rendern.



Um etwas ins Detail zu gehen: Alle 3D-Welten werden mittels Gittermodellen modelliert und mit Texturen überzogen. Je einfacher ein Gittermodell, desto schneller sind die Polygone berechnet und dargestellt. Je detaillierter die Grafik aber sein soll, desto aufwendiger und rechenintensiver wird die ganze Angelegenheit.

Anhand der im folgenden abgebildeten Burg lässt sich der Aufwand für eine reale Spielszene gut abschätzen. Dass diese aufwendigen Berechnungen nur noch von einem darauf spezialisierten Chip erledigt werden können, leuchtet wohl ein.



Um die ganze Sache noch etwas komplizierter zu machen, werden 3 Koordinatensysteme für die Grafikberechnung verwendet. Jedes System hat sich für eine bestimmte Aufgabe besonders gut bewährt.

- World Space
  - Szenerie
- Eye Space
  - Lighting & Culling
- Screen Space
  - Framebuffer

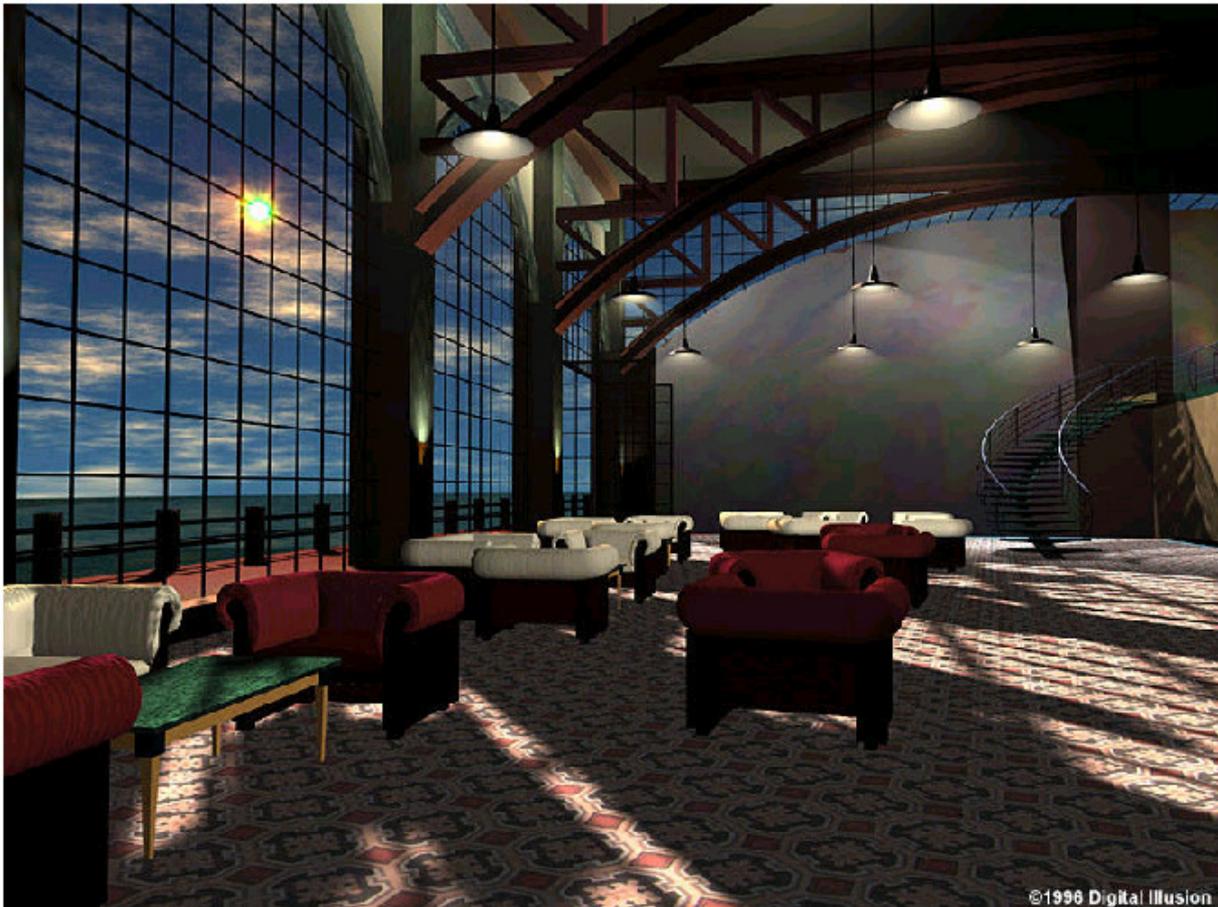
Diese unterschiedlichen Systeme bedeuten aber nochmals zusätzlichen Aufwand, um zwischen den Systemen hin und her zu rechnen.

Die folgende Abbildung zeigt die Entwicklung der letzten Jahre. Es ist gut zu erkennen, dass immer mehr Aufgaben in den Grafikprozessor verlagert wurden, nachdem dieser hardwaremässig für die speziellen Aufgaben optimiert wurde.

Application tasks (move objects according to application, move/aim camera)	CPU	CPU	CPU	CPU	3D Application and API
Scene level calculations (object level culling, select detail level, create object mesh)	CPU	CPU	CPU	CPU	
 Transform	CPU	CPU	CPU	GPU	3D Graphics Pipeline
 Lighting	CPU	CPU	CPU	GPU	
 Triangle Setup and Clipping	CPU	Graphics Processor	Graphics Processor	GPU	
 Rendering	Graphics Processor	Graphics Processor	Graphics Processor	GPU	
	1996	1997	1998	1999	

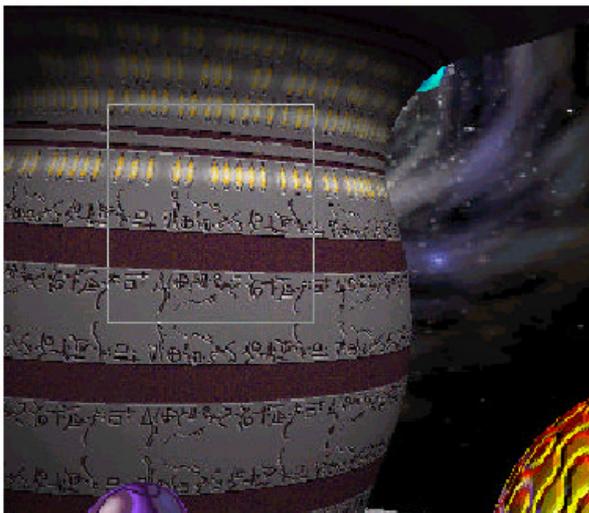
Mit der Entlastung des Hauptprozessors steht jetzt wieder mehr Systemleistung für die KI und andere wichtige Aufgaben zu Verfügung.

Neben der Transformation muss auch die Lighting Engine hervorgehoben werden. Eine virtuelle Realität kann erst als solche bezeichnet werden, wenn neben Farben und detaillierten Gegenständen auch Licht und Schatten verwendet werden. Das menschliche Auge erkennt Helligkeitsänderungen viel besser als Farbänderungen. Deshalb wirkt eine beleuchtete Szene viel eindrucksvoller und hinterlässt mehr Informationen als die gleiche Szenerie ohne Beleuchtung.

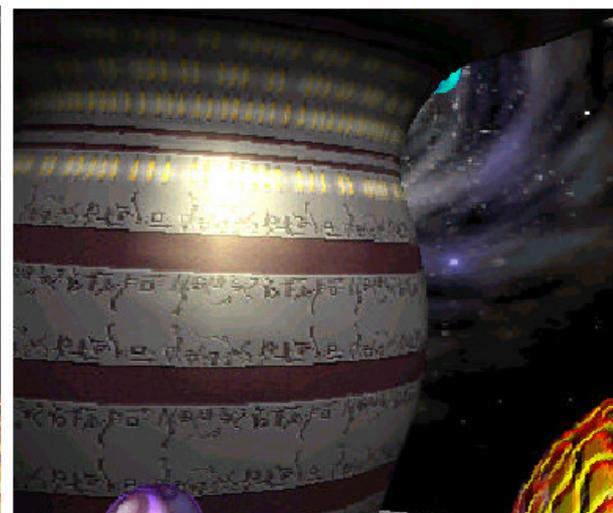


In der PC-Grafik wird eine realistische Beleuchtung mittels zwei Arten von Lichtquellen erreicht. Die erste ist das diffuse Licht, also Sonnenlicht mit parallel einfallenden Strahlen. Mit einem Spotlicht aus einer punktförmigen Quelle werden Lampen und andere Lichtquellen modelliert, welche sich auf den unterschiedlichen Materialien unterschiedlich spiegeln.

Auch die dafür nötigen Berechnungen werden heute von den guten Chips, genauer der Lighting Engine erledigt.



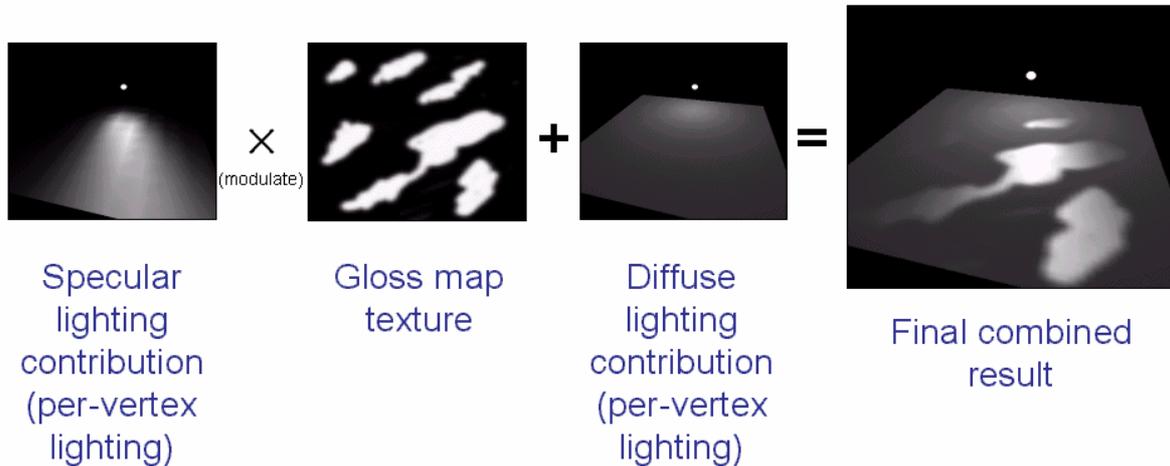
**Diffuse Lighting Only**



**Diffuse and Specular  
Lighting**

Wie uns ein reflektierendes Licht unterschiedliche Materialien vorgaukeln kann, ist anhand der beiden Bilder gut zu erkennen. Im linken Bild ist die Szene nur mit diffusem Licht berechnet. In der rechten Szene wurde ein standortabhängiges Spotlicht zugefügt.

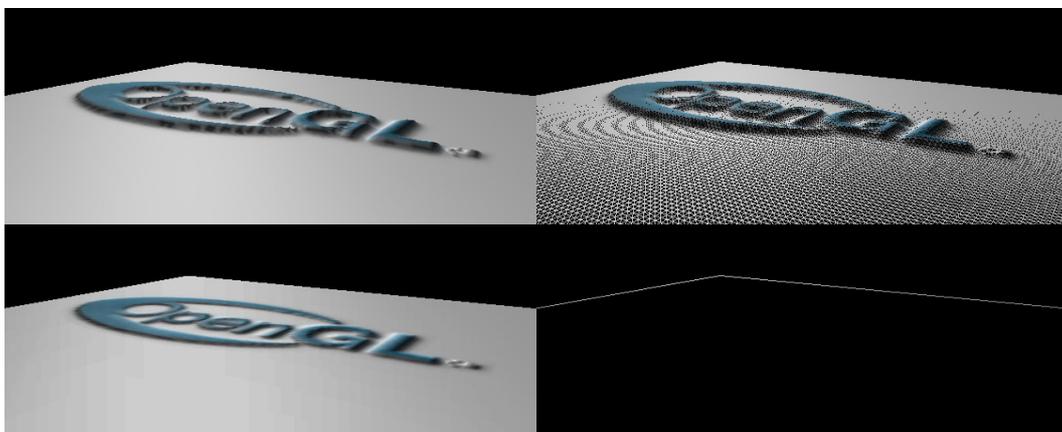
Nebenbei möchte ich noch darauf hinweisen, dass diese Methode relativ neu ist. Eine Alternative zur Vertex- oder PerPixel-Beleuchtung ist das Verwenden von Lichttexturen.



Hier sehen wir eine Kombination beider Verfahren. Zuerst wird ein Spotlicht berechnet und mit einer Lichttextur kombiniert. So erhalten die weißen Flecken einen Helligkeitsverlauf. Diese Flecken werden nun auf einem mit diffusem Licht erhellen Boden aufgezogen. In einer 3D-Applikation würde vorher noch der Boden mit einer Textur beklebt. So hätten wir also pro berechnetem Pixel ein Spot- und ein Diffuslicht, sowie eine Licht- und eine Bodentextur. Damit dies in einer vernünftigen Zeit erledigt werden kann, sind mehrere Textur-Stufen in der Renderpipeline notwendig.

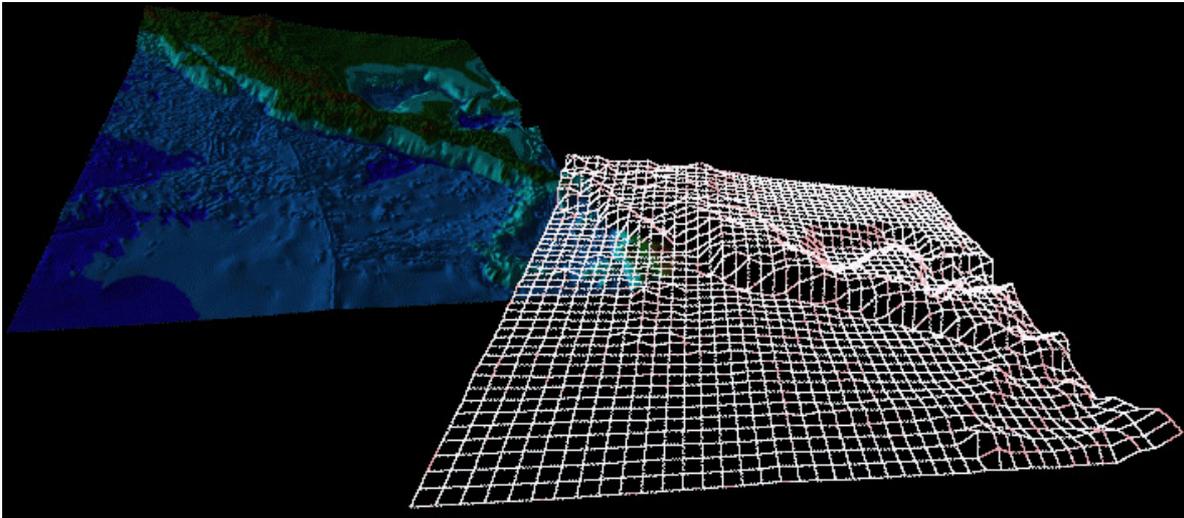
#### 4.3. Bump Mapping

Mit Bump Mapping lassen sich Lichteffekte erzielen, die normalerweise nur mit viel Aufwand zu realisieren wären.



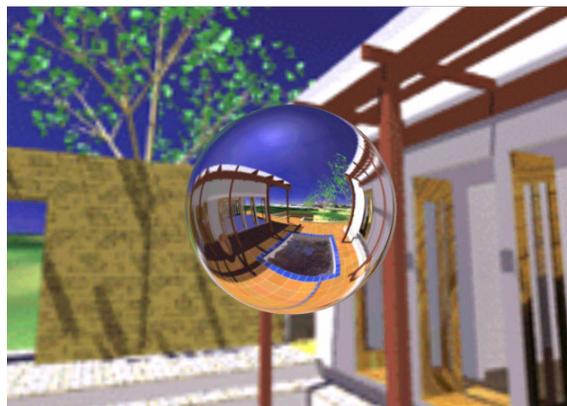
In der oberen Hälfte sehen wir das OpenGL-Logo, welches mit einem dichten Gitternetz modelliert und mit der Lighting Engine beleuchtet wird. Unten ist das

mittels Bump Mapping erreichte, zwar flache, aber genauso beleuchtete Logo dargestellt. Hierzu wurden nur ein Polygon und ein paar Texturen verwendet. So kann eine Realtime-Schattenberechnung erreicht werden, ohne die T&L-Engine unnötig zu belasten.



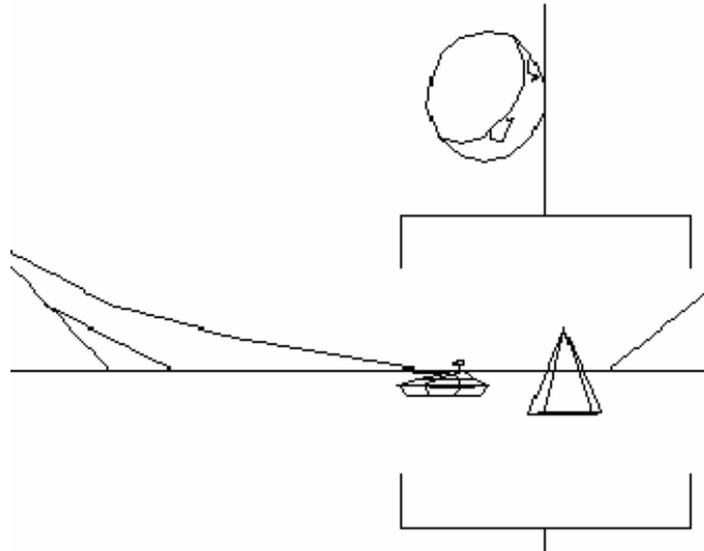
In einer ganzen 3D-Umgebung ist es beeindruckend, welche Details mittels Bump Mapping in eine Szene eingebracht werden können.

Ebenfalls mittels Bump Mapping können Texturen erstellt werden, welche die Umgebung reflektieren. Dies ist jedoch sehr aufwendig und es werden momentan 3 verschiedene Verfahren verwendet. Näher möchte ich nicht darauf eingehen, mehr Informationen zu diesem Thema sind im 3D-Glossar zu finden.



#### 4.4. Anti-Aliasing

Einer meiner Lieblingseffekte ist das Anti-Aliasing oder auch Kantenglättung genannte Verfahren. Dieser Effekt ist nicht programmabhängig, sondern kann über den Treiber aktiviert werden.



Sicher kennen Sie ein altes Spiel oder Pixel-Zeichnungsprogramm, welches bei leicht geneigten Kanten unschöne Treppeneffekte produziert. Dies ist ein noch heute auftretendes Problem im Spielsektor.

Die Chip-Hersteller waren sich bis jetzt noch nicht einig, wie die Kantenglättung erreicht werden soll:

- 3Dfx verwendet das normale Anzeigebild und verschiebt jeweils eine Kopie um weniger als ein Pixel (Sub-Pixel). Anschliessend wird daraus das fertige Bild durch Mittelwerte errechnet. Dies geschieht im Chip selbst, da diese Funktion schon implementiert wurde.
- Die anderen Hersteller müssen diese Funktion softwaremässig anbieten. Dazu wird die Szenerie in mindestens doppelter Auflösung gerendert und anschliessend auf die Anzeigeauflösung heruntergerechnet. Dieser Vorgang ist aber langsamer als der von 3Dfx verwendete Weg.

Vielleicht wird ja der nächste nVidia-Chip diese Funktion auch implementiert haben, da nun nVidia die Rechte am 3Dfx-Verfahren besitzt. Die Qualität ist jedenfalls schon jetzt beeindruckend, wie an dem Spielausschnitt aus Heavy Metal F.A.K.K.2 ersichtlich ist. Links ist das geglättete, rechts das Originalbild ohne Anti-Aliasing zu sehen.



Für meine Karte, eine GeForce256, und einem Pentium III-667 verursacht jedoch schon die niedrigste Kantenglättungsstufe einen zu grossen Leistungsverlust und ein flüssiges Spielen ist nicht mehr möglich.

Deshalb ist die Kantenglättung nur den Spitzen-PCs mit schneller Grafikkarte vorbehalten.

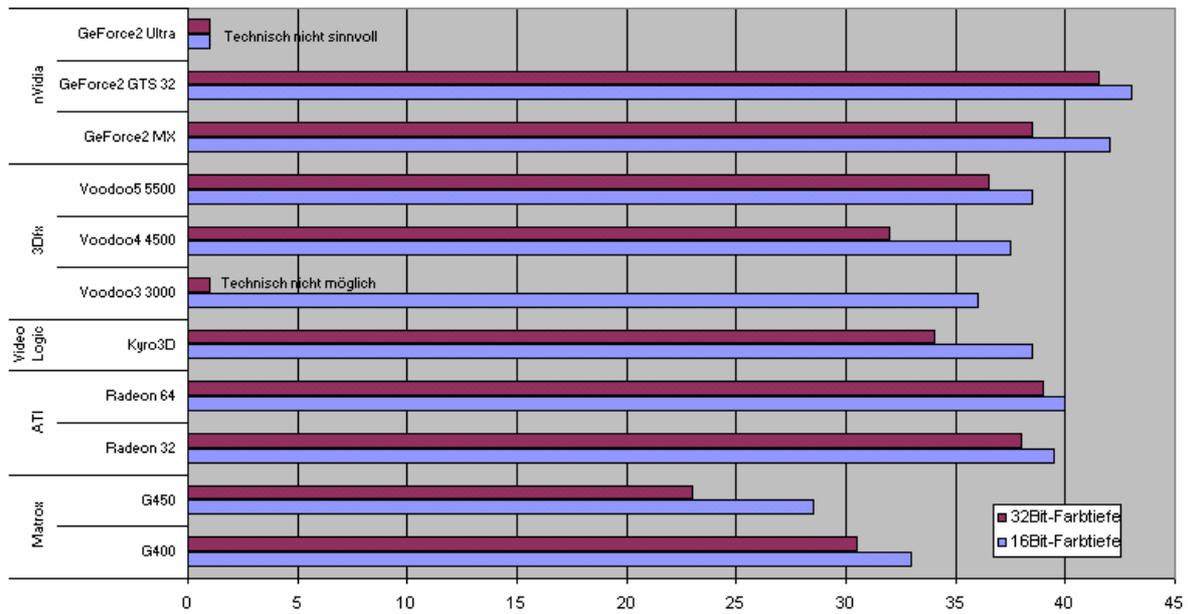
5. Schluss  
5.1. Benchmarks

Wie schlagen sich nun die einzelnen Chips im direkten Vergleich?

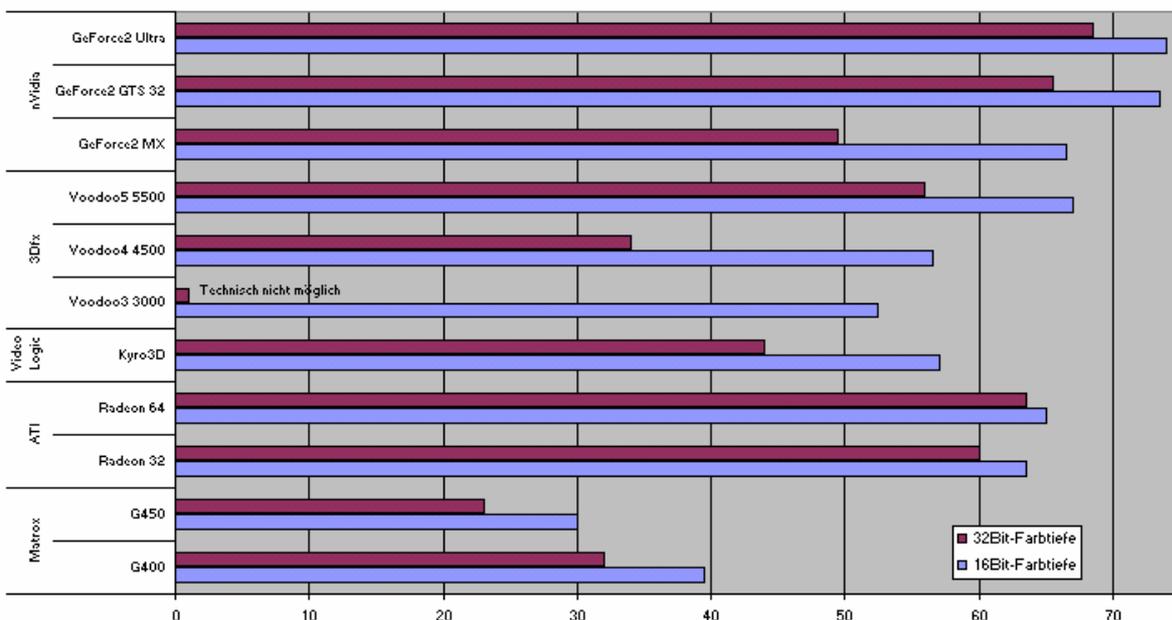
Da die Leistung der Chips stark vom Prozessor abhängt, unterteile ich die Benchmark-Grafik in zwei Bereiche. Für die Framerate wurde je ein OpenGL- und ein DirectX-Spiel verwendet und die jeweiligen Frameraten gemittelt. Wie sich die Karten mit speziell auf OpenGL-, DirectX- oder Glide-optimierten Spielen verhalten, lassen ich aus Platz- und Zeitgründen weg.

Grundsätzlich geht man davon aus, dass 30 Frames pro Sekunde (fps) das Minimum für Computeranimationen sind. Je nach Anwendung oder Spiel können aber auch erst 60 fps ein flüssiges Bild erzeugen.

Frameraten: PIII-500, 1024x768



Frameraten: Athlon-1000, 1024x768



Auffällig ist, dass kostengünstigere Chips wie GeForce2 MX, Voodoo4 4500 und Kyro3D im 32bit-Modus nicht gleich skalieren können, wie die schnellen und damit auch teureren Chips.

Auch ist zu sehen, dass Matrox, zumindest im 3D-Bereich, nicht mithalten kann.

## 5.2. Empfehlung

Aufgrund dieser Benchmarks und aus weiteren markttechnischen Überlegungen mache ich folgende Kaufempfehlungen:

Eine Aufrüstung der Grafikkarte ist für alle Besitzer einer TNT2 oder älteren Karte zu empfehlen. Abhängig von der CPU sind folgende Chips zu bevorzugen:

- ab (PIII 400 MHz): nVidia GeForce2 MX
- ab (PIII 600 MHz): nVidia GeForce2 GTS
- ab (1 GHz): nVidia GeForce2 Pro / Ultra

Wie schon in der Einleitung erwähnt, sind sehr viele Karten mit nVidia-Chips im Handel, welche sich oft nur in der Ausstattung unterscheiden. So sieht für mich eine optimale Karte aus:



## 5.3. Schlusswort

Es wird weiterhin spannend bleiben, da zumindest von nVidia im Jahr 2001 gleich zwei Chipgenerationen erwartet werden und ATI bestimmt auch seinen Beitrag leisten wird.

Beide Hersteller sind übrigens an Gamekonsolen beteiligt, welche Ende 2001 auf den Markt kommen sollen. Also ist auch von dieser Seite ein Schub für die Entwicklung neuer Technologien zu erwarten.

## 5.4. Schluss Aus Ende

So bleibt mir abschliessend nur noch eines zu sagen:

**Lasst die Spiele  
beginnen!**

## 3D-Glossar

### 3D

(Abk. für drei dimensional) Um dreidimensionale (Höhe, Breite, Tiefe) Objekte auf dem 2-dimensionalen Computerbildschirm (Scheibe) möglichst real darstellen zu können, bedarf es einiger z.T. rechenaufwendigen Berechnungen wie z.B. Lichtbrechungen etc..

### 3D-Beschleuniger

Im wesentlichen eine Karte (zusätzliches Teil, das in den Computer gesteckt wird), welche 3D-Berechnungen ausführt. Meist auch gleich zusammen mit der Grafikkarte erhältlich.

### 3D-Chip

Mikroprozessor, der dem Hauptprozessor viele Aufgaben zur Berechnung 3-dimensionaler Szenarien abnimmt. Schattierungen, Transparenzeffekte und Texturen werden beispielsweise von diesem Chip berechnet.

### 3D-Chipsatz

Kombination von mehreren 3D-Chips (z.B einer für Rendering, ein weiterer für z-Buffervergleich)

### 3D-Engine

Aufbau und mathematische Verfahren innerhalb eines Programms zur Berechnung von 3D-Welten.

### 3D Now!

Eine MMX-ähnliche Erweiterung des Befehlssatzes von x86-Prozessoren, welche vor allem 3D-Anwendungen beschleunigen soll. Die 3D Now!-Technologie stammt von AMD.

### 3D-Pipeline

Kette von Arbeitsschritten, bestehend aus Tessellation, geometrischer Transformation und Rendering. Am Ende steht die Darstellung einer imaginären 3D Umgebung auf der 2D-Oberfläche des Monitors.

### 3DRAM

RAM-Entwicklung von Mitsubishi. 3DRAM's sind RAM-Bausteine, mit integrierter Logik, die einige gebräuchliche OpenGL-3D-Operationen (z-Buffervergleich, Blendig von Texturen) direkt im Speicher ausführen. Durch die enorm hohe Bandbreite im innern des Speichers (bis 100GB/s) sind enorm schnelle 3D-Berechnungen möglich! Bis vor kurzem fand man diesen Speichertypus nur in den UltraSpac Hochleistungsgrafikrechner von Sun.

### Add on-Grafikkarte

Spezielle Grafikkarte die nur für bestimmte Bereiche einsetzbar ist (z.B. 3D, TV, etc.). Neben der Add on-Karte ist immer noch eine herkömmliche Grafikkarte erforderlich (z.T. mit mindestens 2 MB Grafikkartenspeicher).

### AGP

(Abk. für Accelerated Graphics Port) Eigentlich kein neuer BUS, sondern nur ein neuer Steckplatz für 3D-Grafikarten - doch irgendwie doch ein neuer BUS, muss doch die Architektur des PCI-Busses abgeändert werden. Entwickelt wurde das ganze von Intel. Durch den direkten Anschluss an den Hauptspeicher kann der Engpass, der früher oder später beim PCI-Bus entstanden wäre, beseitigt werden. Der AGP wurde auch als erweiterte Version AGP 4X mit Bandbreiten jenseits von 1GByte/s produziert. Für AGP-basierende Karten benötigt man auf jeden Fall ein modernes Mainboard (z.B. Intel 440LX/BX) Motherboard. Ausserdem ist der AGP-Bus mit 66 MHz anstatt mit 33 MHz getaktet.

### Aliasing

(Treppeneffekt) Bei der Darstellung von schrägen Kanten und Rundungen bilden sich (v.a. in tieferen Auflösungen) Treppchen, d.h. gezackte Übergänge.

### Anisotropic Filtering

Die ist eine neue Filtertechnik für Texturen, damit diese auch unter flachen Winkeln anschaulich bleiben und Texte darauf lesbar bleiben. U.a. unterstützt der PowerVR2 diese Eigenschaft sowie der RivaTNT und ATI Radeon. Anisotropic Filtering benötigt 16 (!) Texturspeicherzugriffe!

### Anti-Aliasing

Unter Berücksichtigung der benachbarten Pixel (Bildpunkte) in einem Bild werden Kanten geglättet. Es wird aus den 4 benachbarten Bildpunkten ein neuer Farbwert ermittelt (Interpolation). Dieser Effekt unterbindet den Treppcheneffekt bei schrägen Linien. Bei manchen Grafikkarten (z.B. 3Dfx) wird dieser Effekt dadurch erreicht, dass die Pixel am Rand der Kante einen Transparenzwert erhalten.

### Alpha-Blending

("per - Pixel - Effect") Effekt, der 3D-Objekten einen bestimmten Alpha-Wert zuordnet, der dann das Objekt durchsichtig erscheinen lässt. Dieser Effekt kann prima eingesetzt werden um Wasser, Glasflächen, Feuer, Rauch oder Nebel zu simulieren.

**Alpha-Kanal**

Neben den Farbwerten (rot, grün, blau) ist der Alpha-Kanal für die Durchsichtigkeit des Bildpunktes zuständig. Bei einem 8-bit Alpha Kanal können 256 verschiedene Stufen von Transparenz (Durchsichtigkeit) zugeordnet werden. z.B. Wert 0: völlig durchsichtig, Wert 255: undurchsichtig

**Alpha Stipple**

Emulationsmethode für "echtes" Blending, welche von einigen Grafikchips verwendet wird.

**Ambient Light**

Die Helligkeit, welche ohne spezielle Lichtquellen erreicht wird. Direkte Sonnenstrahlung wird ebenfalls ausgeschlossen. Ambient Light besteht nur aus indirekten Lichtstrahlen und ist an einem sonnigen Tag natürlich grösser an einem Regentag.

**API**

Abk. für Application Programming Interface; Ein API ist eine Schnittstelle zwischen Hard- und Software, an der sich die Programmierer orientieren, d.h. für die sie Ihre Spiele schreiben. DirectX ist beispielsweise so eine Schnittstelle. Das Programm kommuniziert nicht direkt mit der Hardware, sondern mit dem API, dieses leitet dann die Signale optimiert weiter zur Hardware.

**Athmospheric Effects**

Nebel und Tiefenunschärfe sind Beispiele für "Stimmungseffekte"

**Artefakte**

Bildfehler, welche entweder durch Ungenauigkeiten in der Soft-/ Hardwareberechnung, den Treibern oder aufgrund von Hardwarefehlern auftreten können.

**Backface Culling**

Es werden aus jeder Perspektive die verdeckten Flächen berechnet. Diese müssen nicht weiter berücksichtigt werden.

**Benchmark**

Leistungsangabe bei Computern, im Grafikbereich meist mit frames per second (fps, Bilder pro Sekunde)

**Bilineares Filtering**

Nähert man sich einer Textur, bilden sich Blöcke. Mit dem bilineareren Filtering werden weiche Übergänge berechnet. Durch Aliasing verursachtes Flimmern wird ebenfalls durch Filtering beseitigt.

**Blending**

Die Farbwerte zweier Pixel werden miteinander vermischt.

**Bump Mapping**

Bump-Mapping bezeichnet eine Reihe von Verfahren, mit denen dreidimensionale Oberflächendetails mit Hilfe von Multitexturing simuliert werden, ohne die Geometrie des Objektes zu verändern oder zusätzliche Polygone einzusetzen. Dieser Effekt kann mit drei verschiedenen Verfahren erreicht werden, von denen jedes für einen bestimmten Anwendungsfall geeignet ist: Emboss, Dot Product 3 und Environment Mapped Bump Mapping (EMBM)

**Clipping**

Methode, um alle momentan nicht sichtbaren Polygone/Objekte eines 3D-Bildes zu ermitteln und abzuschneiden. Diese werden dann gar nicht gerendert. Zum Hardware-Clipping erfordert der 3D-Chip lediglich die Fensterkoordinaten.

**Color Key**

Im Gegensatz zu Alpha Blending werden mit Color Key nicht stufenlose Transparenzwerte zugeordnet, sondern nur die Werte 1 oder 0, was für transparent oder intransparent steht. Das Fehlen dieser Funktion bei manchen Chips führt am Anfang zu vielen Bildfehlern (schwarze Stellen) bei den Spielen.

**CPU**

Central Processing Unit oder Hauptprozessor. Verschiedene Hersteller liefern CPU's, so beispielsweise Motorola, IBM, Intel, IDT und AMD. Einige dieser CPU's sind ebenfalls mit rudimentären 3D-Funktionen ausgestattet (siehe MMX, MMX2, 3DNow!).

**Cubic Environment Mapping**

Das Cubic Environment Mapping ist das flexibelste und gleichzeitig komplexeste Environment Mapping-Verfahren. Es verwendet sechs Environment-Maps, die jeweils eine Seite eines Würfels darstellen, in dessen Zentrum sich der Betrachter befindet. Die verwendeten Spiegelt Texturen sind einfache Quadrate und daher leicht in Echtzeit zu erstellen und zu aktualisieren. Das Verfahren ist ebenfalls unabhängig von der Position des Betrachters, so dass die Spiegelt Texturen nur aktualisiert werden müssen, wenn die Szene sich ändert. Der Nachteil ist der hohe Speicherbedarf für die sechs Spiegelt Texturen. Jedenfalls ermöglicht es schöne Effekte, die vorher undenkbar waren - So können Sie beispielsweise sehen, wie Ihr eigenes Spiegelbild über ein Objekt wandert.

**Culling**

Siehe Backface Culling

### Depth Cueing

Je weiter weg sich ein Objekt befindet, umso undeutlicher wird es dargestellt. Die Objekte verlaufen mit diesem Effekt am Horizont. Im Gegensatz zum Fogging werden die Pixel anstatt mit weissen mit dunklen Pixeln geblendet. Dadurch wird u.a. der "Aufbau-Effekt" (plötzliches Auftauchen von Objekten) verdeckt, der durch das Clipping entsteht.

### Direct 3D

Teil des Treiberespanns DirectX von Microsoft.

### Dithering

Verfahren, um Bilder, die mit einer Farbtiefe von 24bit (True Color) dargestellt werden, in möglichst guter Qualität mit einer geringeren Farbtiefe (z.B. 8-bit = 256 Farben =  $2^8$  Farben) darzustellen. Da die mathematischen Berechnungen teilweise sehr kompliziert sind, ist hier Hardwareunterstützung äusserst sinnvoll.

### Dot Product 3

Dot Product 3 ist ein genaueres und flexibleres Bump-Mapping-Verfahren. Es arbeitet mit einer speziellen Bump-Map-Textur. Zu jedem Pixel der Textur existiert ein 3D-Vektor, der die Neigung der Oberfläche darstellt. Zu jedem Pixel der Bump-Map wird dann ein weiterer Vektor erzeugt, der auf die Lichtquelle zeigt. Anhand des Skalarprodukts (Dot Product) dieser beiden Vektoren wird dann bestimmt, in welche Richtung das Licht von der Oberfläche reflektiert wird. Mit dieser Methode erhält man eine pixelgenaue Bump-Map, durch die die Oberfläche bei Lichteinfall sehr detailliert erscheint.

### Double Buffering

Verhindert durch Synchronisation ständiges Flackern (Tearing) des Bildes, da das Bild in einen Zwischenspeicher (Back Buffer) geschrieben wird und erst auf den Bildschirm (Front Buffer) gelangt, wenn es fertig ist (Page Flipping). Währenddem wird das vorher gerenderte Bild angezeigt. Mit dieser Methode geht aber ca. 25 % der Performance verloren, dafür hat man konstantere, aber tiefere Framezahlen.

### DRAM

Abk. für Dynamic RAM. Mittlerweile veraltete Form von RAM; Diese Art von RAM verliert mit der Zeit die gespeicherten Informationen, wenn die Kondensatoren nicht immer wieder mit Strom versorgt werden. Dieser Vorgang benötigt jedoch Zeit, so dass die Leistung zurückgeht.

### Dual-Paraboloid Environment Mapping

Das Dual-Paraboloid Environment Mapping ist komplexer als das Spherical-Verfahren und verwendet zwei Texturen als Environment-Maps, von denen eine die Umgebung vor dem Objekt und die andere die Umgebung hinter dem Objekt darstellt. Die Texturen sind quadratisch, werden aber mathematisch in die Form einer Halbkugel umgewandelt. Der Vorteil des Dual-Paraboloid Environment Mapping ist seine Unabhängigkeit vom Standort des Betrachters. Daher müssen die Spiegeltexturen nicht neu erstellt werden, wenn der Betrachter sich bewegt, wie dies beim Spherical Environment Mapping der Fall ist. Stattdessen werden die Texturkoordinaten aktualisiert und die Textur neu aufgetragen. Der Nachteil besteht darin, dass die Texturen aufgrund der aufwendigen Umwandlung nicht so schnell erzeugt und aktualisiert werden können, wie dies beim Cubic Environment Mapping der Fall ist. Das Dual Paraboloid-Verfahren benötigt weniger Speicherplatz als das Cubic-Verfahren und eignet sich daher gut für komplexe Reflexionen, die nicht dynamisch aktualisiert werden müssen.

### EDO-RAM

Abk. für Extended Data Output Random Access Memory (Hyper-Page-Mode); Die Bandbreite bei EDO-RAM beträgt maximal 400MB/s. Der Vorteil an EDO liegt darin, dass die zuletzt benötigten Daten beim Anfordern der neuen Daten nicht gelöscht werden, sondern erhalten bleiben. Dieser Vorteil wirkt sich jedoch nur aus, wenn mehrere nacheinander folgende Lesezugriffe folgen, und es sich um ähnliche Daten handelt. Bei der Bilderzeugung folgen meist zusammenhängende Speicherbereiche, wodurch sich der Einsatz im Grafikkartenbereich durchaus lohnt. EDO-RAM als Arbeitsspeicher in einem Cache-System bringt jedoch lediglich einen Leistungszuwachs von 3-5 Prozent.

### Environment Mapping

Environment Mapping verwendet Texturen, in denen sich die Umgebung spiegelt. Mit diesen Texturen kann jeder Oberfläche ein realistischer Spiegeleffekt verliehen werden. Es gibt mehrere Variationen dieser Technik. OpenGL unterstützt Spherical-, Dual-Paraboloid- und Cubic-Environment-Mapping, während DirectX 7 nur das Cubic-Verfahren bietet.

### Farbtiefe

Jedem Pixel kann eine gewisse Grösse an Speicher zugeordnet werden. Bei einer Speichergrösse von 8 bit sind  $2^8=256$  Farbwerte möglich. Im High Color Modus sind 16 bit =  $2^{16}=65536$  Farbwerte möglich und im sogenannten True-Color-Modus sind  $2^{24}=16,8$  Millionen Farben möglich.

### Fogging, Fog

Das überblenden weit entfernter Pixel (Bildpunkte) mit Grautönen nennt man Fogging, da damit ein Nebel-effekt erzeugt werden kann und die Tiefenwirkung verstärkt wird. Dadurch wird u.a. der "Aufbau-Effekt" (plötzliches Auftauchen von Objekten) verdeckt, der durch das Clipping entsteht.

## Flat-Shading

Jedes Dreieck hat eine eigene Farbe. So lassen sich rudimentäre Schatteneffekte bilden. State-of-Art in der Spielbranche bis ca. 1990.

## FPS

(Frames per Second); Bilder pro Sekunde, Das menschliche Auge verarbeitet pro Sekunde ca. 15 Bilder. Um den Eindruck einer völlig flüssig bewegten Szenerie zu bilden, sind jedoch ca. 30 Bilder pro Sekunde nötig. Viele Benchmarkprogramme für Spiele, Video etc. geben die Leistung Ihres PC-Systems in fps an.

## FPU

Floating Point Unit oder Fließkommazahleneinheit. Das ist jener Teil eines Chips/einer CPU, welcher sich um die Berechnung von Fließkommazahlen kümmert. Die FPU ist für 3D-Berechnungen sehr wichtig, viel wichtiger als die Integer-Einheit, welche nur mit ganzen Zahlen umgehen kann, da für die 3D-Darstellung höhere Genauigkeiten erforderlich sind und viel mit trigonometrischen Funktionen gearbeitet wird.

## Frame Buffer

Grafikspeicher, darin wird das Bild aufgebaut, dass später auf dem Bildschirm erscheint.

## Full Screen

Vollbildschirm. Einige 3D-Chips von 3Dfx (Voodoo Graphics und Voodoo 2) konnten nur im Vollbildmodus arbeiten, so dass keine Darstellung im Windows- oder Linux-Fenster möglich war. Alle anderen Chips beherrschen jedoch 3D-Darstellungen im Fenster.

## Geometrische Transformation

Berechnung der Position der 3D-Objekte im Raum, relativ zur Position des Betrachters.

## Geometrieprozessor

Ein Chip, der den Hauptprozessor von den Aufgaben der Tessellation und von Lichtberechnungen entlastet. Meistens (wenn überhaupt) in den 3D-Chip integriert. Er übernimmt die Aufgaben, die sonst die FPU des Prozessors leistet (Fließkommaberechnungen).

## Glide

Rendering-API von 3Dfx zur optimalen Ausnutzung des Voodoo-Chips.

## Gouraud-Shading

Ein mathematisches Verfahren des Mathematikers Henri Gouraud zur feineren Farbgebung innerhalb eines Dreiecks. Die Dreiecke erhalten einen Eckfarbenwert und es wird von Ecke zu Ecke ein Farbverlauf berechnet. Dies ermöglicht recht gute Schatteneffekte und lässt ein Objekt rund bzw. gekrümmt erscheinen, obwohl es facettenartig zusammengesetzt ist. Heute zum Teil noch gebräuchlich in der Spielindustrie in Zusammenhang mit Texture-Mapping.

## HAL

Abk. für Hardware Abstraction Layer, ein Teil eines API's. Im Falle von Direct3D ist HAL ein virtueller 3D-Beschleuniger, mit allen 3D-Funktionen, welche das aktuelle System bieten kann. HAL ist quasi die Schnittstelle zwischen Programm und 3D Hardware. Das Programm wendet sich also an HAL.

## HEL

Hardware Emulation Layer: Wiederum ein Teil eines API's. Im Falle von Direct3D gaukelt HEL der Software die nicht vorhandene 3D Hardware vor. Der Programmierer muss sich also nicht darum kümmern, ob das System seine Funktionen Hardwarebeschleunigt oder nicht. HEL lässt die Funktion dann per Software emulieren, falls Direct3D diese kennt.

## Hidden Surface Removal

Nach dem Z-Buffering werden die unsichtbaren Objekte/Polygone aus dem Bild entfernt; da dies geschieht, bevor das Bild gerendert wird, wird danach keine Prozessorkapazität vergeudet.

## Interpolation

Aus zwei oder mehreren benachbarten Punkten oder auch Flächen wird ein mathematischer Mittelwert errechnet. Durch diverse Abstufungen werden feinere Übergänge erzeugt.

## Lens Flare

Effekt, der entsteht, wenn man eine Kamera auf eine helle Lichtquelle richtet.

## LOD

Abk. für Level of Detail, Begriff in Zusammenhang mit MIP-Mapping

**MMX**

(MultiMedia Extension) Befehlserweiterung um 57 Instruktionen für alle Pentium und Nachfolge-Prozessoren, die vor allem auf Grafik, Sound und Videoanwendungen zugeschnitten ist. Die Prozessoren erhalten mit der MMX-Technik acht 64 Bit grosse MMX-Register dazu. Zudem gibt es vier neue Datentypen: Packed byte, Packed word, Packed doubleword und Quadword. Die MM-Erweiterung sollte eigentlich auch 3D-Spiele, welche direkt für MMX programmiert sind, beschleunigen, was sich allerdings kaum bewahrheitet hat. Die MMX-Technologie befindet sich (seit 8.1.97) in allen neuen Pentium™ Prozessoren und seit Mitte '97 auch in allen Konkurrenzprodukten (K6, C6, 6x86MX).

**MMX2**

(MultiMedia Extension 2) Technologie von Intel, welche weitere Befehle spendiert. Nachdem MMX sein Augenmerk auf 2D- und Video legte, will Intel nun verstärkt FPU/3D-Instruktionen implementieren um so der K6's 3DNow-Technologie die Stirn zu bieten.

**MIP-Mapping**

(MIP Abk. für **M**ultum **I**n **P**arvum; viele unter gleichen); Um bei Annäherung an ein Objekt mehr Details erkennen zu können, werden diesem Objekt mehrere Texturen zugeordnet. Je näher der Betrachter kommt, umso mehr Details werden sichtbar. Hochauflösende Texturen werden in der Nähe verwendet, niedrigauflösende für entfernte Objekte. In Zusammenhang mit tri-linearem Filtering sehr wirkungsvoll.

**Motion Capturing**

realistische Simulation von bewegten Lebewesen.

**OpenGL**

Die Standardbibliothek zur 3D-Programmierung.

**Overlay Planes**

Speicherbereiche auf der Grafikkarte, in denen Buttons, Mauszeiger und andere Grafikobjekte abgelegt werden, welche oft benötigt werden und/oder über den Bildschirm gezogen werden.

**Perspective Correction**

Die Verzerrung einer 3D-Textur, die bei der perspektivischen Darstellung entsteht, wird mit diesem Effekt ausgeglichen!

**Pixel**

(Abk.: Picture Element) Bildpunkt, jeder Bildpunkt hat eine bestimmte Koordinate und einen bestimmten Farbwert. Der Punkt „oben-links“ hat die Koordinate (0,0), derjenige „unten-rechts“ am Bildschirm je nach Auflösung verschieden (z.B. 768,1024).

**Pixeldurchsatz**

Wert, der Aufschluss über die Organisation des Speicherzugriffs auf Textur- und Bildspeicher. Gemessen wird dieser am besten mit grossen Dreiecken und möglichst vielen aktivierten Effekten.

**Phong-Shading**

Neben den Eckfarbwerten (Gouraud-Shading) wird zudem ein Normalenvektor für die Reflektionsstärke der Oberfläche angegeben, der eine noch realistischere Darstellung ermöglicht! Phong-Shading wurde nach dem Mathematiker Phong Bui-Tong benannt. Echtzeiteinsatz sehr rechenaufwendig. Bei Spielen nur mit Hardwarebeschleunigern sinnvoll.

**Polygon**

engl.: Vieleck, Im 3D-Bereich jedoch meistens ein Dreieck; Alle Objekte einer 3D-Welt setzen sich aus Polygonen zusammen.

**Polygondurchsatz**

wird mit vielen kleinen Dreiecken gemessen; Misst vor allem die Leistung des Geometrieprozessors (z-Buffering, MIP-Mapping). Ist dieser nicht vorhanden, drückt das den Wert, da dann alles der Hauptprozessor erledigen muss.

**Point Sampling**

Technik für das Texture Mapping, welche weniger aufwendig als bilineares Texture Mapping ist. Wird eine Textur verändert, werden die Pixel aus der ursprünglichen Textur errechnet, aber nur nach den "welcher Pixel ist in meiner Umgebung am nächsten" Methode. Der Klötzcheneffekt wird kaum beseitigt.

**Pre-Rendered**

Vorberechnete Animation, die nur noch abgespielt werden muss. Keine Echtzeit-Berechnung, daher auch kein freier Bewegungsraum.

**Ray-Tracing**

Strahlungsverfolgung zur realistischen Berechnung der Lichteffekte in einem Raum, inkl. Spiegelungen und Schattierungen.

**RAMDAC**

Abk. für Random Access Memory Digital/Analog Converter; Endstufe der Grafikkarte, die das Signal in für den Monitor verwendbare Signale umwandelt. Heutzutage meistens in den Grafikchip integriert.

**Real-time**

(engl. Echtzeit), Spiele werden meistens in Echtzeit berechnet, da dies eine maximale Bewegungsfreiheit für den Spieler garantiert. Echtzeit meint, dass die 3D-Szenerie gerade jetzt, in diesem Moment vom Computer berechnet wird, und nicht als Film auf der Festplatte oder CD gespeichert ist und nur noch abgespielt werden muss.

**Rendering/Rendern**

Für die Darstellung einer 3D-Szenerie erforderlicher Rechenprozess. Position und Farbe eines Punktes im 3D-Raum werden ermittelt. Die Tiefe wird im Z-Buffer abgespeichert, Farbe und Position (Länge, Breite) im Framebuffer. Danach werden die verdeckten Werte im Z-Buffer ermittelt (Z-Buffering).

**RDRAM**

Rambus DRAM, Leistungsfähiges RAM mit einer Bandbreite von über 1 GB, speziell geeignet für Grafikanwendungen.

**Shading**

Allgemein: Schattierung von 3D-Objekten; Gekrümmte Flächen werden in viele kleine Dreiecke aufgeteilt!

**Shutter-Brille**

Dank Shutter-Brillen ist es möglich, echte Räumlichkeit zu erzeugen. Dies ist auf einem Monitor (2D!) nicht möglich. Shutter-Brillen funktionieren nach einem einfachen Prinzip: Jedes unserer Augen hat einen leicht anderen Blickwinkel. Beide Augen zusammen ermöglichen uns die räumliche Wahrnehmung. Ausserdem kann das menschliche Auge nur ca. 20 Bilder pro Sekunde voneinander unterscheiden. Shutter-Brillen nutzen diese "Features" des Menschen aus und verschliessen, synchronisiert mit dem Monitor, jeweils kurzzeitig die Sicht für eines der Augen und stellen dann für das andere Auge den entsprechenden Blickwinkel auf dem Monitor dar. Abwechselnd erhalten nun beide Augen jeweils ein anderes Bild auf dem Monitor. Dies hat allerdings einen Nachteil. Da die Bilder nun sehr häufig gewechselt werden müssen (ca. 30x/s), sinkt die Bildwiederholfrequenz um die Hälfte. Bei 120 Hz welche die Grafikkarte liefert, bleiben jeweils pro Auge noch 60 Hz übrig, das ist weniger als die minimal ergonomischen 75 Hz. Um "flimmerfrei" mit 3D-Shutterbrillen spielen zu können, muss die Grafikkarte in der entsprechenden Auflösung schon 150 Hz liefern und der Monitor die Frequenz verkraften können.

**SLI**

Abk. Scanline Interleaving; Durch das Parallelschalten von mehreren Karten bzw. Chips/Chipsatz wird die Rechenarbeit auf mehrere Chips verteilt. Die eine Karte übernimmt dann das Rendering der einen Hälfte der Bildschirmzeilen, den Rest übernimmt die andere. Entweder werden zwei Grafikkarten mit einem Kabel verbunden oder zwei Chipsätze auf einer Karte untergebracht. Bis heute nutzt dieses Verfahren nur die Voodoo-Chipsätze.

**Specular Highlights**

Glanzlichter, welche im Zusammenhang mit Sonnenstrahlen bei metallischen Oberflächen oder bei Kamerablenden entstehen. Sehr rechenintensiv.

**Spherical Environment Mapping**

Das Spherical Environment Mapping verwendet eine einzige Textur als Spiegeltextur. Ein Beispiel zur Erklärung des Verfahrens: Stellen Sie sich vor, Sie stehen still und bewegen nur den Kopf. Je nachdem, in welche Richtung Sie schauen, sehen Sie ein anderes Bild. Stellen Sie sich nun vor, Ihr Sichtbereich würde auf eine Kugel aufgemalt, in deren Zentrum Sie stehen. Wenn dieses Bild flach ausgebreitet würde, wäre es wie eine Spherical-Spiegeltextur oder eine Environment-Map. Diese Environment-Map kann um dreidimensionale Objekte gelegt werden, um realistische Reflexionen zu erzeugen.

**Stencil Buffer**

Dient zur Begrenzung des Zeichenbereichs. Dank diesem Buffer muss nicht dauernd die Software die Lage der Objekte zum Fensterrand überprüfen.

**SDRAM/SGRAM**

Schneller Speichertyp mit synchronem Zugriff. SDRAM's besitzen 16 Datenkreuzungen, SGRAM's deren 32 und zudem spezielle Schreibbefehle (Block-, Mask-Write).

**Tesselation**

Erster Arbeitsschritt beim Berechnen eines 3D-Bildes: Unterteilung der Objekte in Polygone (Dreiecke). Dabei erhält jeder Punkt des Dreiecks Raumkoordinaten, Farbwerte und evtl. noch Transparenzwerte.

**Texel**

Texturierter Pixel.

**Textur**

Bild/Muster, das über ein 2D- oder 3D-Objekt gelegt wird und es somit realistischer aussehen lässt.

**Texture Caching**

Wie beim Prozessor werden häufig gebrauchte Texturen in einem schnellen Zwischenspeicher abgelegt, auf den der 3D-Chip direkt Zugriff hat.

**Texture Composing**

Mehrere Texturen werden übereinandergelagert und miteinander vermischt.

**Texture Mapping**

Dieser Effekt versieht ein Objekt mit einem Bild und berechnet auch die perspektivische Verzerrung. Sehr rechenaufwendig! Seit 1991 üblich bei Computerspielen.

**Trilineares Filtering**

Ähnlich Bilineares Filtering, jedoch einiges rechenaufwendiger (ca. 8 mal aufwendiger als bilinear), da der Mittelwert von zwei bilinear gefilterten Pixeln genommen werden muss! Effekt wird unter anderem für das MIP-Mapping verwendet.

**Vertex**

Ein Punkt im 3-dimensionalen Raum

**VR**

(Abk. für Virtual Reality); künstliche, mit dem Computer generierte Wirklichkeit.

**Wireframe**

Ein 3D-Objekt wird dabei nur mit den Kanten als Drahtgitter dargestellt, Flächen werden keine angezeigt und somit wird auch im Hintergrund nichts verdeckt. Gängige 3D-Beschleuniger sind nicht auf Wireframe-Darstellung optimiert! Drahtgitter/Wireframedarstellungen waren vor ca. 20 Jahren das einzige, was in Echtzeit berechnet werden konnte.

**Z-Buffer/-ing**

Die Tiefeninformation eines Objektes muss bei fast allen 3D-Grafikkarten in einem speziellen Speicher gespeichert werden. Beim Z-Buffering vergleicht die Grafikkarte ständig die Tiefenwerte und entscheidet dann, welches Objekt für den Betrachter oder die Lichtquelle sichtbar ist oder nicht.

**Z-Sort/-ing**

Beim Z-Sorting wird einfach alles gerendert; zuerst die hintersten Objekte, diese werden dann von den vorderen überdeckt; viel rechenaufwendiger als Hidden Surface Removal.

## Quellennachweis

### Textquellen ganzer Vortrag

nVidia; <http://www.nvidia.com>

ATI; <http://www.ati.com>

3D Concept; <http://www.3dconcept.ch>

ComputerInfo; <http://www.computerinfo.ch>

3Dfx; <http://www.3dfx.com> (Homepage seit Dez. 2000 geschlossen)

PCGames; <http://www.pcgames.de>

PCGames Hardware; <http://www.pcgameshardware.de>

PCPlayer; <http://www.pcplayer.de>

### Folie 4 – Falschdeklaration

microspot; Weihnachtskatalog 2000; <http://www.microspot.ch>

### Folie 7 – Der Untergang

PCGames Hardware 02/2001 – Der Untergang von 3Dfx; <http://www.pcgameshardware.com>

### Folie 15 – Renderpipeline-Texture Units

ATI Chip Technology – Visuelle Details – Aus jedem Pixel das Letzte herausholen; <http://www.ati.de>

### Folie 20 – Fast Writes

nVidia – Technical Brief - AGP 4X With Fast Writes – A Unique NVIDIA GeForce 256 Feature;

<http://www.nvidia.com>

### Folie 24-26 – Transform & Lighting

nVidia – Introducing the World's First GPU: GeForce 256; <http://www.nvidia.com>

nVidia – NVIDIA GeForce 256 - Transform and Lighting; <http://www.nvidia.com>

nVidia – GeForce2 GTS Second Generation GPU; <http://www.nvidia.com>

### Folie 28-31 – Transform & Lighting

nVidia – Technical Brief - Transform and Lighting; <http://www.nvidia.com>

### Folie 32 – Transform & Lighting

GDC 2000 Programming Session: Computations for Hardware Lighting and Shading;

<http://www.nvidia.com>

### Folie 34-35 – Bump Mapping

GDC Tutorial: Advanced OpenGL Game Development - A Practical and Robust Bump-mapping

Technique for Today's GPUs; <http://www.nvidia.com>

### Folie 36 – Environment Mapping

Bubble-Demo; <http://www.nvidia.com>

### Folie 38 – Anti-Aliasing

Screenshot aus Multiple Arcade Machine Emulator

### Folie 40 – Anti-Aliasing

Screenshots aus Heavy Metal F.A.K.K.2

### Folie 44 – Schlusswort

Screenshots aus Rune, Drakan, Heavy Metal F.A.K.K.2, Midtown Madness 2, Sacrifice, Black&White, Stupid Invaders, Giants

### Anhang 1 – 3D-Glossar

3D Concept – Begriffe rund um 3D; <http://www.3dconcept.ch>

ATI Chip Technology – Visuelle Details – Aus jedem Pixel das Letzte herausholen; <http://www.ati.de>

## **InternetLinks**

### **Online-Magazine für 3D-Grafik**

**RivaStation**

<http://www.rivastation.com/>

**3D Concept**

<http://www.3dconcept.ch/>

### **Tweaking & Tools**

**GeForce Tweak Utility 2.5.1**

<http://www.guru3d.com/geforcetweakutility/download.html>

**Tweak PC Hardware Guide**

<http://www.tweakpc.de/>

**HZ Tool**

<http://hem.spray.se/doxx/>

### **Benchmark & Technologiedemos**

**3DMark 2000**

<http://gamershq.madonion.com/products/3dmark2000/>

**TechnoDemos GeForce2 GTS**

<http://www.nvidia.de/Products/geforce2gts.nsf/action.html>

### **Diverses**

**Preisvergleiche von PC-Hardware**

<http://www.tech1.de/pricewatch/>

**Pressemitteilung & FAQ von nVidia zum Kauf von 3Dfx**

[http://www.nvidia.de/News/Pages.nsf/pages/pm\\_121500faq](http://www.nvidia.de/News/Pages.nsf/pages/pm_121500faq)