Projektarbeit

Breakout mit C++ und Opengl mit einem Aspket einer Game-Engine

Marco Pattke

015

Inhalt

**Es wurden keine Einträge für das Inhaltsverzeichnis gefunden.**

# Einleitung

Das Computerspiel Beakout wurde ursprünglich 1976 von Steve Wozniak als Arcade-Spiel rein in Hardware realisiert. Das Spiel wurde jedoch seit dem auf Atari und Apple portiert und es gibt zahlreiche Klone, die direkt Online, Auf dem Smartphone etc. spielbar sind.

Ziel des Spiels ist es, mit Hilfe des vom Spieler kontrollierten Schlägers den Ball so abzulenken, dass der Ball "Mauersteine" trifft um diese zu zerstören. Erst nach dem alle Steine zerstört wurden, gilt das Level als gewonnen. Sollte der Ball nicht vom Spieler abgelenkt werden können und er berührt den unteren Bildschirm-Rand, ist das Spiel verloren.

Ziel dieser Projektarbeit ist es, dieses Spiel, angelehnt an das originale, mit der Programmiersprache C++ und der Grafikbibliothek OpenGL zu realisieren. Auf Tonausgabe wird keinen Wert gelegt.

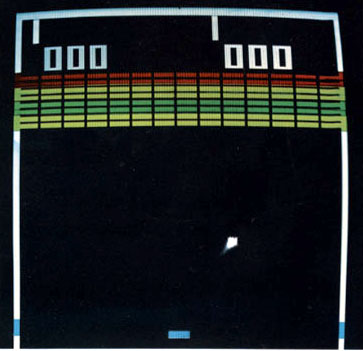


Abbildung 1 - Bild vom ursprünglichen Breakout auf dem Atari

# Grundlagen

## Was ist OpenGL?

OpenGL (Open Graphics Library) ist eine plattformunabhängige Grafikbibliothek, die inzwischen zu einem weltweiten Standard für 3D-Grafikprogrammierung und zu einer der meist genutzten 3D APIs geworden ist. Sie wird sehr vielfältig verwendet: In der Industrie, in Forschungszentren, in der Spieleprogrammierung und in der Lehre.

Die Plattformunabhängige wird dadurch erreicht, dass OpenGL tatsächlich nur die Spezifikation einer Bibliothek ist, für die verschiedenste Implantierungen existieren, teilweise direkt von Graphikkarten unterstutzt. Es gibt OpenGL Bibliotheken in fast allen erdenkbaren Programmiersprachen, z.B. für C, C++, Java, C# und sogar für Google GO.

OpenGL wurde 1992 von Silicon Graphics entwickelt, wird jedoch seit Juli 2006 von der Khronos Group verwaltet. Die Khronos Group ist eine Vereinigung mehrerer Unternehmen, die sich für die Erstellung und Verwaltung offener Standards im Multimediabereich auf einer Vielzahl von Plattformen und Geraten einsetzt. Die Aktuelle Version von OpenGL ist die Version 4.5 (seit 11. August 2014). Für OpenGL wird keine Lizenz benötigt, da seit 2009 alle Teile von OpenGL als freie Software lizensiert wurde.

## Was leistet OpenGL?

OpenGL stellt eine universelle Methode dar, relativ unkomplizierte komplexe 3D-Grafiken zu erzeugen. Durch die gezielte Anordnung einfacher Grafikelemente werden komplexe Körper gebildet, welche 3D-Szenen bilden. Primitiven sind Punkte, Linien und Flächen (Polygone). Aufbauend auf diesen Grundgeometrien hält OpenGL auch Grundkörper wie Kugeln, Zylinder oder gekrümmte Oberflächen in gesonderten Bibliotheken bereit.

OpenGL wurde als Zustandsmaschine entworfen. Attribute eines Objektes, wie z.B. Farben beim Zeichnen, bleiben bis zur Änderung erhalten. Anbieter wie z.B. Grafikkartenhersteller können diese Zustandsmaschine um eigene Zustände erweitern und Funktionen ausbauen.

# Game-Engine

Eine Game-Engine ist eine Sammlung von Technologien und Hilfsmitteln, die die Erstellung

von Spielen in großem Maße unterstützt und erleichtert. Eine Engine umfasst dazu eine große

Menge von Funktionen, die von nahezu allen Spielen verwendet werden, und keinen spezi-

fischen Spielinhalt. Eine Game-Engine wird auch des Öfteren als „Betriebssystem für Spiele“

bezeichnet. Es existieren Engines, die sich nur für die Erstellung von Spielen eines bestimmten

Genres eignen und dafür optimiert sind, wie z.B. einige Engines für Spiele aus der

Ego-Perspektive, die nur in Gebäudekomplexen angesiedelte Szenarien und keine weitläufigen

Außenlandschaften zulassen, dafür aber über sehr ausgefeilte Beleuchtungsmodelle verfügen.

Andere Engines hingegen versuchen eine möglichst breite Bandbreite an Funktionalität anzubieten,

um die auf ihrer Basis erstellten Spiele so wenig wie möglich im Genre und in den

Möglichkeiten einzuschränken.

Als Aspekt einer Game-Engine habe ich in meiner Arbeit ist Austauschbarkeit der Render-Engine implantiert. Dadurch kann man zwischen OpenGL und DirectX wählen.

# Umsetzung