

Universität Innsbruck
Fakultät für Informatik
Proseminar: Wissenschaftliches Arbeiten
Leitung: assoz. Prof. Dr. MOSER Georg



Spieleprogrammierung

Juni 2013

Matthias WANNER (4. Semester)
Schneeberggasse 49a
6020 Innsbruck
Email: matthias.wanner@student.uibk.ac.at

Inhaltsverzeichnis

1	Einleitung	1
2	Spielprogrammierung - Die einzelnen Bereiche	2
2.1	Game Design	2
2.2	Entwicklung eines Prototypen	2
2.3	Testen	3
2.4	Fertigstellung	3
2.5	Wartung	3
2.6	Game Engine	4

1 Einleitung

Am Anfang eines Computerspiels steht eine Idee. Um die Idee in ein erfolgreiches Spiel umsetzen zu können, braucht es gewisse Phasen, die durchlaufen werden müssen. Früher war die Komplexität wesentlich geringer und ein Spiel ist in kürzerer Zeit produziert worden. Heutzutage sitzen bis zu 30 Programmierer an einem Bereich in der Entwicklung und am Ende liegt der Schnitt der Fertigstellung, bis zu drei Jahre. Allgemein kann man die Entwicklung in folgende Teile gliedern.

- Game Design
- Grafik
- Level Design
- Spieleprogrammierung
- Sound

Das **Game Design** kann man als Regieführung sehen. Hier wird beschlossen, wie Charaktere aussehen, die Story verläuft, oder an was für Richtlinien sich das Spiel halten soll. Die **Grafik** ist für das Erstellen von Texturen und Modellieren von Welten, Figuren zuständig. Das **Leveldesign** geht noch etwas genauer auf die Spielewelten ein. Es wird außerdem eine Atmosphäre für den Spieler erstellt. Damit das Spiel auch akustisch gut ankommt, wird im Bereich des **Sounds** nicht gespart. Nicht selten werden ganze Orchester verwendet, um wie beim Film eigene Szenen zu vertonen. Die wohl wichtigste Phase ist die **Spieleprogrammierung** selbst. Sie ist auch der Kleister der die verschiedenen Phasen zusammenhält. Die Programmierung haucht dem Spiel das Leben ein. Mit Hilfe einer Game Engine¹ können die einzelnen Teile zusammengeführt werden. Solche Engines werden von Spieleentwicklern verwendet und immer auf das jeweilige Spiel angepasst. In der Entwicklung gibt es viele Arbeitsfelder, eines davon sind die Programmierer. Sie sind für das Schreiben des Source Codes verantwortlich. Programmiert wird in viele verschiedenen Sprachen, jedoch eignen sich C, bzw. C++² am besten und sind damit auch die am weit verbreitetsten.[2] Für die Spiele im 21. Jahrhundert gibt es mittlerweile viele Plattformen³ die in Frage kommen können. Je nach Auftrag und Portierbarkeit der Engine, können Spiele für mehrere Plattformen, oder auch nur für eine einzelne erstellt werden. Die folgenden Seiten beschreiben nun den Teil der Spieleprogrammierung, damit aus dem Code ein fertiges Spiel wird und nicht der Charakter sich einfach in Luft auflöst.

¹Wird später noch genauer behandelt.

²C ist besonders Hardware nah. C++ ist wegen der Objekt orientierten Programmierung sehr wichtig.

³Plattformen sind die Endgeräte auf denen das Spiel gespielt wird. z.B.: PC, Xbox, Playstation

2 Spieleprogrammierung - Die einzelnen Bereiche

2.1 Game Design

Wenn man den Beruf eines Game Designer ausübt, so hat man die volle Verantwortung über das Projekt. Das Game Design bezieht sich auf all die wichtigen Dinge die ein Spiel ausmacht. Das Genre in dem es platziert wird, die Story die hinter dem Spiel steckt, die Vorgabe des Gameplays, die Regeln an die es sich halten soll, sowie die gesamte Umgebung. Der Game Designer ist vergleichbar mit einem Produzenten, oder Regisseur eines Films - er hat die Zügel in der Hand. Im Game Design sind vor allem die Richtlinien und der Weg festgelegt in die das Spiel gehen soll. Es wird ein Spielansatzdokument¹ angelegt, welches wie ein Leitfaden für die Entwicklung bereitsteht.[1] In großen Entwicklerfirmen, wie Electronic Arts, Ubisoft, oder auch , Activision Games, werden die einzelnen Bereich, wie Gameplay, Grafik, Level Design, etc. von mehreren Lead Designer verwaltet und geleitet. Durch eine bessere Strukturierung entsteht eine weniger fehleranfällige Produktion und die Entwicklungsdauer wird verkürzt. Im Game Design muss oft abgewogen werden, ob Spiel-Elemente verwirklichtbar, ob die Hardware-Anforderungen möglich sind und ganz wichtig, ob das Budget ausreicht für die Entwicklung. Zwischen all diesen Dingen muss das Game Desing, bzw. der Game Designer hin und her jonglieren um die goldene Mitte zu finden.

2.2 Entwicklung eines Prototypen

Bevor das Spielansatzdokument überhaupt abgeschlossen ist, fängt man schon an Prototypen zu bauen. Prototypen sind kleine Programme, oder Skizzen die helfen sollen, Eigenschaften des Spiels zu veranschaulichen. An ihnen kann man leicht Spielverläufe ausprobieren. Zum Beispiel muss eine Cutscene² getestet werden. Hinter den Prototypen steckt auch der Versuch eine Idee auszuprobieren und zu testen. Sie sind zur Verwirklichung von Konzepten vorhanden. Außerdem sind Prototypen nicht nur für die Spielfähigkeit zuständig, sondern werden auch erstellt um andere Bereich des Spiels zu überprüfen, wie das Interface, oder die grafische Oberfläche. In der Zeit vor der Produktion werden viele Prototypen erstellt um einige verschiedene Konzepte zu testen. Sei es die Bewegung der Charaktere oder auch nur die Geräuschkulisse beim Gehen eines Spielers. Prototypen sind im Gegensatz zu Testprogrammen, nicht nur auf den Code beschränkt, sie sind auch im Bereich von Design, Sound, Physik und Grafik im Einsatz.

¹Ein Dokument das die Richtlinien und Vorgaben des Kunden bzw. des Game Designer beinhaltet.

²Eine Zwischensequenz, meist in Form eines kurzen Videoclips, in der die Handlung vorangetrieben wird.

2.3 Testen

Früher waren Spiele weniger komplex und es wurde meistens vom gleichen Programmierer getestet. Heutzutage ist dies nicht mehr möglich und große Publisher³ beauftragen ganze Firmen um die Qualitätsbestimmungen zu erfüllen. Einen eigenen Standard in der Spielebranche gibt es nicht und somit testet jede Firma auf ihre eigene Weise. Sobald die erste Zeile Code geschrieben wird fangen die Tests an. Tests sind wichtig um die Stabilität des Codes bzw. des Spiels zu ermitteln. Es werden mögliche Fehler provoziert, um sie zu finden und zu eliminieren. Einen fixen Zeitpunkt, wann getestet wird gibt es nicht. Die Tests laufen immer wieder parallel neben der Produktion mit. Das Arbeitsfeld eines Spiele-Testers beinhaltet den Fund von nicht abgehandelten Fehlern und diese werden gesammelt und dem Entwickler Team übergeben. Sogenannte Bugs⁴ sind in drei bestimmte Kategorien eingeteilt.[3]

- Klasse A : Kritische Fehler, die die unmittelbare Fertigstellung verhindern und das Spiel abstürzen lassen.
- Klasse B : Fehler die überarbeitet werden sollten, jedoch das Spiel ist noch spielbar.
- Klasse C : Mehr ein Vorschlag als ein Fehler. Klein und nicht von großer Wichtigkeit.

Spiele werden vor allem auf die Funktionalität, die Einhaltung der Regeln und Stabilität getestet.

2.4 Fertigstellung

Am Ende einer Produktion, kommt die Fertigstellung des Spiels. Sie beinhaltet das „Aufpolieren“. Nun werden die Alpha und Beta Phasen gestartet, bzw. ausgewertet. In der Spieleentwicklung ist die Alpha und Beta Phase von großer Bedeutung. Hier wird das Spiel, entweder öffentlich zugänglich gemacht, oder nur für eine kleine Auswahl, an potentielle Kunden vergeben. Nach diesen Phasen wird die Software nochmal überarbeitet und die gefundenen Fehler von den Testern behoben. Außerdem werden die Server, bei Spielen die eine Internetverbindung brauchen, ausgelastet und einem größeren Stresstest unterzogen. Nachdem all diese Test zufriedenstellend sind und der Publisher das Ok gibt, wird das Computerspiel an die Händler ausgeliefert. Das Spiel befindet sich dann sogenannten Gold-Status. Also frei von (bekannten) Fehlern.

2.5 Wartung

Sobald die ersten Spiele verkauft worden sind, sammeln die Programmierer alle möglichen Fehlerberichte von Konsumenten und filtern diese nach Wichtigkeit. Wenn genügend Material zusammengekommen ist, wird versucht einen Patch mit der Fehlerabhandlung zu erstellen. Der Patch beinhaltet Fehlerbehebungen, jedoch kann er auch Spielerweiterungen im Bereich der Handlung, oder auch mögliche neue Inhalte wie Waffen, Charaktere, oder neue Karten haben.

³Ein Publisher ist die Firma die das Spiel in der Öffentlichkeit Vertritt und vermarktet. Publisher beauftragen Entwicklerfirmen für die Erstellung von Spielen. Beispiel: Electronic Arts

⁴Englisch für Fehler

2.6 Game Engine

Softwareentwicklungsfirmen, die sich auf die Herstellung von Computerspielen spezialisiert haben, verwenden Game Engines die stets wiederverwendet werden. Grob kann man sagen, dass die Game Engine aus der Grafik Engine, der Physike Engine und der Sound Engine besteht. Künstliche Intelligenz und andere Themen werden auch noch in eine Engine zusammengefasst. Am Ende ist die Game Engine ein kompaktes Paket Code, das möglichst effizient und speichersparend arbeiten soll. Die **Grafik Engine** ist vor allem für das Laden von Texturen und die Renderings zuständig. Sie muss Unmengen an Daten in kürzester Zeit verarbeiten und sollte möglichst nah an die Hardware anliegen. Die sogenannten API's⁵ sind die Schnittstellen zwischen Programmierer und der Hardware. Die zwei bedeutendsten Grafik API's sind DirectX und OpenGL.

Um Spiele von der Bewegung her möglichst realistisch darzustellen, braucht es die **Physik Engine**. Sie wird dazu verwendet um Personen, wie im echten Leben gehen zu lassen, oder lässt Gegenstände naturgetreu fallen. Mit dem Ragdoll-Verfahren⁶ kann man genauestens die Fallbewegung einer Person berechnen und implementieren. Am Anfang als die Ragdoll-Engine zum ersten mal eingesetzt worden ist, war die zu realistische Erscheinung sehr ungewohnt und nicht sehr beliebt. Heutzutage ist sie kaum noch wegzudenken, da sie vor allem im Bereich von Action-Kriegsspielen von großer Bedeutung ist.

Damit auch der akustische Part des Menschen stimuliert wird, braucht ein Spiel eine **Sound Engine**. Mit Surroundsound kann der Spieler in Actionspielen akustisch die Position eines Gegners bestimmen und hat dadurch einen klaren Vorteil, jedoch ist hierfür auch die richtige Ausrüstung vonnöten. Sei es ein gutes Headset, oder eine Soundkarte.

Im Großen und Ganzen ist die Spieleprogrammierung ein Teilbereich, im Entwicklungsprozess eines Spiels. Um in dieser Branche beruflich tätig zu werden, ist es möglich an gewissen Universitäten bzw. Hochschulen Game Design als Kurs zu belegen. Damit hat man den Einstieg in die Spieleprogrammierung geschafft. Zum Schluss ist zu sagen, dass das Thema der Spieleprogrammierung in dieser Seminararbeit sehr verkürzt dargestellt worden ist. Diese Thema umfasst noch viel mehr interessantere Bereiche der Programmierung, von den genutzten Tools, bis hin zu Entwicklung von Autonomer Intelligenz.

⁵API steht für application programming interface. Also die Anwenngensprogrammierung dient als Schnittstelle.

⁶Ragdollphysik: Vom englischen „Ragdoll“, bedeutet Stoffpuppe und wird dazu verwendet möglichst genaue menschliche Bewegungen zu simulieren.

Literaturverzeichnis

- [1] Digital Worlds Interactive Media and Game Design. The process of camge creation and the game design document. *Operations Research*, 2008. Zugriff: 4. Juni. 2013.
- [2] GameDevelop.eu Christoph P. Einführung in die spieleentwicklung. *Operations Research*, 2011. Zugriff: 3. Juni. 2013.
- [3] the free encyclopedia Wikipedia. Game testing, 2013. Zugriff: 4. Juni. 2013.