

PREPRINT – Endfassung mit Abbildungen veröffentlicht:

Loebel, J.-M.: *Interaktion mit Games mittels Emulation im musealen Kontext*.

In: Deutsches Filminstitut - DIF e.V. (Hg.), Lenhardt, E.; Rauscher, A. (Red.): Film und Games: Ein Wechselspiel. Berlin: Bertz und Fischer, 2015. ISBN: 978-3-86505-241-4, ISBN (engl. Fassung): 978-3-86505-242-1, S. 230-233.

Interaktion mit Games mittels Emulation im musealen Kontext

Jens-Martin Loebel, bitGilde IT Solutions UG (haftungsbeschränkt)

<loebel@bitgilde.de>

Digitale Objekte sind die Güter des Informationszeitalters und bilden wertvolle kulturelle und wissenschaftliche Ressourcen. Sie sind damit Teil unseres digitalen Kulturerbes, dessen Bewahrung und Zugänglichkeit eine der großen Herausforderungen an unsere Gesellschaft darstellt. Wie können digitale Objekte angemessen präsentiert und zugänglich gemacht werden? Wie lässt sich unser digitales Kulturerbe dauerhaft und adäquat bewahren? Von der Gesellschaft für Informatik jüngst zu einer Grand Challenge¹ erklärt, ist diese stark interdisziplinäre Aufgabe Gegenstand zahlreicher Forschungs- und Bewahrungsprojekte.

Die Erhaltung digitaler Daten ist dabei kein einmaliger Vorgang, sondern ein dauerhafter Prozess. So ist beispielsweise die Haltbarkeit digitaler Datenträger deutlich kürzer als die analoger Medien. Datenbestände müssen in regelmäßigen Abständen kopiert und auf die jeweils neueste Generation von Speichermedien überführt werden. Zusätzlich sind die zugrunde liegende Hard- und Software – Betriebssysteme, Treiber, Abspielprogramme sowie Datenformate – einem ständigen Wandel unterworfen. Hinzu kommt eine stetig wachsende Menge von zumeist proprietären Datei- und Datenformaten, die das weitaus größere Problem darstellt. Digitale Daten müssen interpretiert werden, damit sie für Menschen in einer sinnlich wahrnehmbaren Form rezipiert werden können: Daten- und Dateiformate, die als gespeicherter Bitstrom vorliegen, werden mittels eines Softwareprogramms und gegebenenfalls weiterer Komponenten dargestellt. Es müssen Wege gefunden werden, sämtliche Softwarekomponenten – Objekt, Abspielprogramm, Zusatzprogramme, Treiber und Betriebssystem – über Generationen von Rechner- und Softwaresystemen hinweg dauerhaft benutzbar zu halten.

Die Bewahrung komplexer und interaktiver digitaler Artefakte wie Computer- und Videospiele birgt dabei zusätzliche Anforderungen aufgrund ihres hohen Grads an Interaktivität und der Vereinigung vieler medialer Elemente, beispielsweise Bild, Ton, Video, 3-D. Zeitgleich rückt die Erhaltung des *Look-and-feel* des Spiels und des Dispositivs aus Soft- und Hardware inklusive der Ein- und Ausgabegeräte, der Rezeptionsumgebung sowie der historischen Aufführpraxis in den Vordergrund.

¹ Vgl. www.gi.de/themen/grand-challenges-der-informatik/digitale-kultur.html [01.02.2015].

Emulation als Bewahrungsstrategie

Die Methode der Emulation hat sich als einzig gangbare Möglichkeit bewährt, um die Eigenschaften dieser Klasse von komplexen digitalen Objekten obsoleter Computersysteme zu erhalten.

Emulation als Softwaretechnik findet in der Informatik breite Anwendung und ist eine gut untersuchte Technik. Ursprünglich in den 1960er Jahren von IBM entwickelt, um alte Programme auf der neu eingeführten und nicht abwärtskompatiblen 360-Rechner-Architektur auszuführen, ermöglichen es Emulatoren, die temporale und technologische Kluft zwischen zwei Rechnersystemen zu überbrücken.² Ein Emulator ist dabei ein spezielles Softwareprogramm, welches Teileigenschaften eines Rechnersystems nachahmt und zur Laufzeit Maschinenbefehle eines Systems in entsprechende semantisch äquivalente Softwarebefehle eines anderen Systems übersetzt. Ein solcher Emulator ist in der Lage, die Software eines Systems A, beispielsweise eines obsoleten Heimcomputers oder Videospiegelgeräts, auf einem System B, zum Beispiel einem aktuellen PC, auszuführen, und erzeugt im Idealfall die gleiche Ausgabe.

Im Spielebereich wird die Entwicklung solcher Emulatorprogramme vornehmlich von einer Community von Enthusiasten angetrieben, die sich über das Internet vernetzt und austauscht mit dem Ziel, Spiele ihrer Kindheit und Jugend und die dazugehörigen Systeme dauerhaft spielbar zu erhalten. Quasi als Kollateralnutzen existieren heutzutage für fast jedes obsolete System ein oder mehrere von Hobbyentwicklern erstellte quelloffene Emulatoren. Die Erhaltung mittels Emulation ist essenziell, um den Charakter komplexer digitaler Objekte zu bewahren und eine authentische Wiedergabe zu ermöglichen.³

Herausforderungen im Ausstellungskontext: Ein Prototyp für das Filmmuseum

Für die Ausstellung *Film und Games. Ein Wechselspiel* wurden Ausstellungsstationen entwickelt, an denen Besucher Spielszenen interaktiv spielen können. Neben der Betrachtung einer vergleichenden Filmsequenz ist die direkte Ansteuerung bestimmter Szenen von Spielen verschiedenen Alters und unterschiedlichster Systeme notwendig. Konzept und technische Herausforderungen wurden in einem

² Vgl. Stuart G. Tucker: Emulation of large systems. In: *Communications of the ACM*, Band 8, Nr. 12, 1965, S. 753–761.

³ Vgl. Jens-Martin Loebel: *Lost in Translation – Leistungsfähigkeit, Einsatz und Grenzen von Emulatoren bei der Langzeitbewahrung digitaler multimedialer Objekte am Beispiel von Computerspielen*. Glückstadt: Verlag Werner Hülsbusch (vwh), 2014. Abrufbar unter: www.translation-gap.de [18. 02. 2015].

ersten Prototyp aus Hard- und Software von der Firma bitGilde IT Solutions UG eruiert.⁴

Das System stützt sich auf eben jene quelloffenen Emulatoren als hochspezialisierte Software zur Aufführung digitaler Spiele. Der Code der Emulatorprogramme wurde dazu um Funktionen erweitert, um den Zustand des emulierten „virtuellen“ Systems automatisiert zu speichern und beim Start wiederherstellen zu können (sogenannte *Save States*). Zur Erzeugung eines *Save State* wird innerhalb des emulierten Systems der aktuelle Zustand des Prozessors und aller Systemkomponenten sowie der Inhalt des System- und Grafikspeichers erfasst und anschließend serialisiert in einer Datei abgespeichert.

Dabei lag das Augenmerk auf der Entwicklung eines automatisierbaren Ladesystems für diese *Save States*, eingebettet in ein Auswahlménü mit Webservice und Skriptsteuerung. Grundgedanke des Prototyps war die Entwicklung eines Frameworks, das die Machbarkeit des Konzepts zeigt und eine ausbaufähige Architektur bereitstellt.

Die Ausführung im musealen Bereich stellt darüber hinaus weitere Anforderungen. So musste darauf geachtet werden, dass eine einfache Wartung und Austauschbarkeit einzelner Komponenten sowie eine Robustheit der Hardware unter Dauerbenutzung möglich sind. Dies wurde durch die Verwendung von Standard-PC-Komponenten und Spielcontrollern erreicht, die für Besucher nicht sichtbar verbaut werden. Daneben konnten durch den Einsatz und die Erweiterung freier Software gleichzeitig die Gesamtkosten weiter minimiert werden. Emulatoren benötigen, um ausgeführt zu werden, prinzipbedingt ein System, das deutlich leistungsfähiger ist als das emulierte Originalsystem. Hier musste abgewogen werden zwischen Leistung und geringer Wärme- und Geräusentwicklung (durch Systemlüfter) der PCs. Um einen konstanten Besucherstrom zu gewährleisten, wird das Spiel durch Magnetsensoren beendet, wenn Besucher den Controller zurück ans Exponat legen. Zudem wird die Spieldauer zeitlich begrenzt.

Das Auswahlménü und die Skriptsteuerung werden als dynamische Webseiten mit Anbindung an einen lokalen Webservice realisiert. Dadurch werden das Design und die Nutzerinteraktion vom Rest des Systems getrennt. Ménü und Design können bei Bedarf ohne besondere technische Kenntnisse angepasst werden.

Ausgeführt wird die Webanwendung auf dem Browser Firefox, der in einen speziellen Kioskmodus für Ausstellungen und öffentliche Plätze versetzt wurde. Dadurch ist es möglich, die Interaktion mit den Besuchern genau festzulegen und das darunter liegende System in der Ausstellung zu verbergen. Es ist geplant, die Ergebnisse und den Prototypen unter einer freien Lizenz zu veröffentlichen, um eine vielfältige Nachnutzung zu ermöglichen.

⁴ www.bitgilde.de [18. 02. 2015].