

# Kinderleicht wie ein Computerspiel?

## Versuch einer Analyse

Sven Türpe

Fraunhofer-Institut für Sichere Telekooperation, Darmstadt

<tuerpe@sit.fhg.de>

### Zusammenfassung

*Dieser Beitrag befaßt sich mit der These, Computerspiele seien aufgrund einfacher Bedienbarkeit ein gutes Vorbild für Anwendungen anderer Art. Er zeigt an Beispielen, wie und warum Grundprinzipien der Softwareergonomie in Computerspielen absichtlich verletzt werden, um diese Spiele zu einer interessanten Beschäftigung zu machen, und zieht daraus Schlußfolgerungen.*

### 1. Intro

Technisch ist in der Welt der virtuellen Realitäten (VR) alles in Butter. Ein- und Ausgabegeräte für VR-Systeme existieren, Algorithmen gibt es in Hülle und Fülle, und die nötige Rechenleistung zur Erzeugung von 3D-Grafik in Echtzeit liefert jeder PC vom Discountr.

Durchgesetzt haben sich virtuelle Umgebungen (virtual environments, VE) als allgemeine Benutzungsschnittstellen jedoch nicht. In einigen Nischen werden VR-Techniken erfolgreich eingesetzt, etwa in Simulatoren zur Pilotenausbildung oder zur anschaulichen Visualisierung von Konstruktionen vor ihrer tatsächlichen Herstellung. Ansonsten existieren sie jedoch hauptsächlich als Forschungsprototypen, beeindruckend bei Vorführungen, aber im Alltag bedeutungslos.

Die Suche nach Gründen muß von den Motivationen und Hoffnungen ausgehen, mit denen VE-Ansätze für die verschiedensten Anwendungen vorgeschlagen wurden. Ein gern angeführtes Argument ist die Ähnlichkeit zu Computerspielen, welche zum einen erfolgreich 3D-Techniken einsetzen und zum anderen als einfach und intuitiv bedienbar, also benutzerfreundlich gelten.

Der vorliegende Beitrag sammelt vorhandenes Wissen und versucht, anhand einer kurzen Analyse gängiger Spieltypen den Geltungsbereich dieser Aussage einzugrenzen. Abschnitt 2 gibt einen Überblick über Com-

puterspiele und erinnert an die Voraussetzungen für die Benutzerfreundlichkeit von Anwendungssoftware. In Abschnitt 3 werden exemplarisch einige Klassen von Computerspielen vorgestellt und analysiert. Der letzte Abschnitt zieht Schlußfolgerungen aus den gemachten Beobachtungen.

### 2. Hintergrund

#### 2.1. Computerspiele

Was ist eigentlich ein Computerspiel? Naheliegende Definitionen, wie etwa »alles, was als solches angeboten wird« oder »Software zum Zeitvertreib« sind leider wenig hilfreich. Auf eine eigene Begriffserklärung wird hier verzichtet; statt dessen sollen Eigenschaften von Computerspielen und Klassifikationskriterien aus vorhandenen Quellen zusammengetragen werden.

Eine kurze Definition des Begriffs *Computerspiel* liefert Drolshagen in einer sozialwissenschaftlichen Arbeit. Er kennzeichnet sie als »*Programme (...), die speziell für spielerische, lustbetonte, und von äußeren Zwecken freie Tätigkeiten geschrieben wurden und auf einem Personalcomputer ausgeführt werden.*« [7]. Wichtig scheint hier die Freiheit von äußeren Zwecken, welche ein fundamentales Unterscheidungsmerkmal zwischen Spiel- und Anwendungssoftware darstellt.

Weitaus ausführlicher hat sich Crawford in der Frühzeit der PCs mit den besonderen Merkmalen von Computerspielen befaßt [4]. Er stellt sie in eine Reihe mit Brett- und Kartenspielen sowie sportlichen Wettkämpfen, unterscheidet sie aber vom kindlichen Spiel, etwa der phantasiereichen Improvisation mit beliebigen Gegenständen.

Crawford macht vier gemeinsame, grundlegende Elemente in allen Spielen aus: *Darstellung, Interaktion, Konflikte und Sicherheit*. Ein Spiel ist nach seiner Auffassung »*a closed formal system that subjectively represents a subset of reality*« [4], wobei die Darstellung keineswegs

exakt und umfassend ist, sondern des Spielers Phantasie durch Vereinfachung anregt und einbezieht. Die Art der *Interaktion* beziehungsweise ihr Stattfinden überhaupt unterscheidet Spiele von Puzzles, Geschichten und Spielzeugen.

*Konflikte* treten in Spielen auf, wenn sich dem Spieler beim Verfolgen seines Ziels – Erreichen einer im formalen System willkürlich festgelegten Gewinnsituation – Hindernisse in den Weg stellen. Im Gegensatz zu Drolshagen, der sich mit Kooperation in Computerspielen befaßt [7], hält Crawford Konflikte für eine notwendige Eigenschaft von Spielen jeglicher Art. Darin, daß solche Konflikte im Spiel, wie auch immer sie (innerhalb des Regelsystems) ausgetragen werden, in der Regel keine negativen Folgen im wahren Leben zeitigen, liegt schließlich die *Sicherheit* von Spielen.

Neben den Eigenschaften von Spielen beleuchtet Crawford auch die Motivationen der Spieler. Für den stärksten Motivator hält er die Freude des Menschen am Lernen (im Sinne von »Neues kennenlernen oder entdecken«). Weiterhin nennt er Anregung der Phantasie, die Möglichkeit, gefahrlos gegen Regeln des täglichen Lebens oder gar physikalische Gesetze zu verstößen, das Sich-selbst-beweisen und Anerkennung finden, Training kognitiver Fähigkeiten und die Funktion von Spielen als soziales »Schmiermittel«.

In einer weiteren, fast zehn Jahre jüngeren Schrift [5] stellt Crawford eine baumartige Klassifikation und Hierarchie der *Entertainment*-Systeme (*Interactive Entertainment*) auf. Sie führt zur Unterscheidung zwischen *interaktiven Geschichten, Toys, Puzzles, Wettbewerben* sowie den *Spielen* im engeren Sinne. Was landläufig als Computerspiel bezeichnet wird, kann in eine jede dieser Klassen fallen. Moderne Unterhaltungssoftware – Crawfords Klassifikation entstand Anfang der neunziger Jahre – kombiniert allerdings oft Merkmale mehrerer Klassen.

Auf klassische Videospiele bezieht sich Belding [2], der eine Liste von ebenfalls vier Merkmalen anbietet. Deinen erstes ist *Anwesenheit (Presence)* des Spielers in der Welt des Spiels. Der Eindruck der Anwesenheit wird erzeugt durch Bewegung und Kollision, konsistente Regeln innerhalb der Spielwelt, Bewegungsfreiheit des Spielers, Echtzeitinteraktion und Hintergrundaktivität, Dreidimensionalität – kurzum, alle Eigenschaften, die für gewöhnlich von einer virtuellen Realität erwartet werden.

Das zweite Merkmal bezeichnet Belding als *Chaos*, worunter er alles Unvorhersehbare, Unerwartete, Überraschende subsumiert. Als drittes Element führt er die *Herausforderung (Challenge)* an. Ein Spiel muß ausreichend schwierig zu meistern sein, sonst wird es schnell langweilig. Schwer zu meistern bedeutet aber nicht notwen-

dig auch schwer zu lernen; ein Spiel kann intuitiv bedienbar sein und dennoch schwierig und herausfordernd. Einstellbare Schwierigkeitsgrade und das in vielen Spielen auftretende Level-Konzept unterstützen die ausreichende Herausforderung bei kleiner Einstieghürde.

Als viertes Merkmal schließlich führt Belding die *Atmosphäre* an, welche durch Grafik, Geräusche, Musik und die gesamte Gestaltung des Spiels vermittelt wird. Sie bildet das künstlerische Element und hilft dabei, den Spieler zu fesseln und ihm ein Spielerlebnis zu vermitteln. Eine Ausprägung dieses Merkmals kann etwa die detailreiche Darstellung in modernen Computerspielen sein.

## 2.2. Software-Ergonomie

Von »ernsthaften« Anwendungen wird Ergonomie, Benutzerfreundlichkeit, Usability verlangt. Diese Forderung hat inzwischen zum Beispiel in Form der deutschen Bildschirmarbeitsverordnung [3] sogar Eingang in die Gesetzgebung gefunden.

Ein Maß für Software-Ergonomie gibt es jedoch nicht, ebensowenig ein Entwicklungsmodell, dessen Befolgung sie garantiert. Statt dessen behilft man sich mit Faustregeln und, zur Beurteilung, mit der Beobachtung von Benutzern in der Hoffnung, dabei Probleme aufzuspüren [10, 11]. Selbst die Liste der Faustregeln variiert von Autor zu Autor. Als notwendige Eigenschaften benutzerfreundlicher Anwendungssoftware werden beispielsweise genannt [9][8][3]:

- Aufgabenangemessenheit, Natürlichkeit  
Das System unterstützt die Erledigung der Arbeitsaufgabe, ohne zu behindern; es bildet ein geeignetes Werkzeug zur Erledigung der Aufgabe. Mit Natürlichkeit ist hier nicht die möglichst detailgenaue Nachbildung von Phänomenen und Sachverhalten der realen Welt gemeint, sondern Natürlichkeit bezüglich der zu bewältigenden Aufgaben.
- Selbstbeschreibung, Transparenz, Unterstützung, Lernbarkeit  
Das System selbst vermittelt dem Benutzer Informationen über Einsatzzweck, Leistungsumfang und Bedienung. Der Benutzer kann den Systemzustand bezüglich seiner Aufgabe jederzeit erkennen. Alle jeweils benötigten Informationen werden angezeigt, einschließlich zum Beispiel Navigationshilfen. Irrelevante Informationen werden verborgen. Eine Online-Hilfe allein genügt zur Selbstbeschreibung nicht.
- Steuerbarkeit, direkte Manipulation [15]  
Der Benutzer bestimmt Geschwindigkeit und Rei-

henfolge der Interaktionshandlungen selbst. Die Arbeit kann jederzeit unterbrochen werden.

- Erwartungskonformität, Konsistenz, Kompatibilität  
Der Benutzer kann vorhandene oder in der bisherigen Benutzung gemachte Erfahrungen weiter verwenden. Ähnliches Dialogverhalten in ähnlichen Dialogsituationen; gleiches in gleichen. Das System benutzt das Vokabular, welches der Benutzer mit der Aufgabe verbindet. Inwieweit Realweltmetaphern hierbei hilfreich sind, ist bislang nicht abschließend geklärt.

- Anpaßbarkeit, Flexibilität

Anpassung an persönliche Kenntnisse, Erfahrungen, Fähigkeiten und Vorlieben des Benutzers. Anpassung von Informationsdarstellungen an momentanes Informationsbedürfnis.

- Fehlerrobustheit, Toleranz

Kritische (besser: alle) Aktionen können rückgängig gemacht werden. Es gibt keine undefinierten Systemzustände. Die Rückkehr aus einem Fehler- in einen Arbeitszustand ist einfach und schnell zu bewerkstelligen.

Solcherlei Regeln greifen freilich zu kurz. Während ihre Mißachtung sicher zu unergonomischer Software führt, ist ihre Anwendung allein kein Garant für Ergonomie. Als weiterer Input sei deshalb hier Normans *The Design of Everyday Things* [12] angeführt, nicht wegen des häufig mißverstandenen [13] Begriffes *Affordance*, sondern wegen des dort erläuterten siebenstufigen Modells der Ausführung von Aufgaben und der dort hervorgehobenen Bedeutung des konzeptionellen Modells, womit das Interface-Design zum Systemdesign wird.

### 3. Beispiele

In diesem Abschnitt wird nun an einigen Beispielen untersucht, wie die Eigenschaften von Computerspielen und die Regeln zur ergonomischen Softwaregestaltung in typischen Spielprogrammen zusammenwirken. Besonderes Augenmerk liegt dabei auf der Frage, ob und wo Abweichungen von den guten Sitten und Gebräuchen des Anwendungs-Designs auftreten.

#### 3.1. 3D-Shooter

##### 3.1.1. Beschreibung

3D-Shooter (auch als 1st-Person-Shooter oder Ego-Shooter bezeichnet) sind die in den letzten Jahren wohl

erfolgreichste Anwendung von VR-Techniken nicht nur im Bereich der Computerspiele. Vertreter dieser Klasse sind beispielsweise die Spiele Doom, Quake und Duke Nukem 3D. Sie entsprechen allesamt Crawfords Definition von *Spielen* im engeren Sinne (*Games*).

Der Spieler wird in eine künstliche, dreidimensionale Welt versetzt, welche er aus der Perspektive einer sich darin bewegenden Spielfigur wahrnimmt. Die Darstellung ist detailreich und umfaßt neben der Grafik auch Töne. Zur Steuerung wird bei neueren Vertretern des Genres meist eine Kombination aus Maus bzw. Joystick und Tastatur benutzt. Primäres Ziel des Spielers ist das Erreichen eines Ausgangs, der zum nächsten Level oder, am Ende des Spiels, zum Spielgewinn führt. Daneben gibt es Mehrspielerzonen, in denen das Ziel zum Beispiel im Erobern der Flagge eines gegnerischen Teams besteht, sowie teilweise strategische Aufgaben. Letztlich geht es jedoch immer darum, einen Weg durch die Spielwelt zu finden und erfolgreich zurückzulegen. Um dieses primäre Ziel zu erreichen, muß der Spieler mehrere Sekundärziele verfolgen.

Die Bewegungsfreiheit des Spielers wird durch die Simulation physikalischer Eigenschaften in der Spielwelt eingeschränkt. Insbesondere sind Wände und andere Objekte undurchsichtig und undurchdringlich, und die Spielfigur kann nur auf geeigneten Oberflächen laufen. Um zum Ziel zu gelangen, muß der Spieler oft erst einige Türen öffnen, wozu entweder mehr oder weniger gut versteckte Schalter oder vorher aufzufindende Schlüssel dienen. Auch getarnte Geheimtüren kommen zuweilen in diesen Spielen vor. Die Suche nach einem Weg sowie nach Mechanismen oder Utensilien zur Bewältigung von Hindernissen bildet also ein sekundäres Ziel des Spielers.

Der Spieler ist nicht allein in seiner Welt. Ihm stellen sich Gegner in den Weg, die entweder vom Computer, oder, im Mehrspielermodus, von anderen menschlichen Spielern geseteuert werden. Diese Gegner sind ebenso wie die gesteuerte Spielfigur bewaffnet. Begegnungen sind nicht zwangsläufig tödlich; jeder Treffer verringert aber die Zahl der »Gesundheitspunkte«. Fällt diese Zahl auf null, ist das Spiel beendet. Überleben ist also das zweite Sekundärziel des Spielers. Im Mehrspielermodus kommt unter Umständen noch die Notwendigkeit hinzu, bei Begegnungen blitzschnell zwischen Freund und Feind zu unterscheiden und richtig zu reagieren.

Die anfänglich schwache Bewaffnung kann mittels aufgefunder Gegenstände verbessert werden, was die Überlebenschancen verbessert. Zudem verbrauchen die Waffen Munition; Ersatz ebenfalls an verschiedenen Stellen der Spielwelt gefunden werden. Ebenso läßt sich die

Gesundheit zum Beispiel mittels aufgefunder »Medikits« oder durch das Trinken von Wasser wiederherstellen. Weitere Gegenstände verleihen besondere Fähigkeiten wie Fliegen, Tauchen oder Unverwundbarkeit. Das dritte Sekundärziel ist das Auffinden solcher Gegenstände. Im Gegensatz zu den oben erwähnten Schlüsseln sind sie nicht grundsätzlich notwendig; der Bedarf oder die Nützlichkeit ergibt sich erst aus der konkreten Spielsituation.

### 3.1.2. Analyse

Die Aspekte der Darstellung und Sicherheit (Crawford) sowie der Anwesenheit des Spielers und der Atmosphäre (Belding) dienen in erster Linie der Motivation des Spielers und werden hier nicht detailliert diskutiert. Interasanter, weil in Beziehung zu den Anforderungen an ergonomische Software stehend, sind Interaktion, Konflikte, Chaos und Herausforderung.

Die Spielwelt ist ein klassisches Labyrinth. Menschen sind, das weiß schon die griechische Mythologie mit ihrem Faden der Ariadne, nicht besonders gut darin, sich in Labyrinthen zurechtzufinden [14]. Unzählige Landmarken und deren Lage zueinander sind im Gedächtnis zu behalten, was bei großer Ähnlichkeit zwischen verschiedenen Teilen des Labyrinths besonders schwierig ist. Das Herumlaufen im Labyrinth hat einen – im Spiel erwünschten – Nebeneffekt: es kostet Zeit proportional zur Entfernung zwischen zwei Punkten. Gelegentlich findet man allerdings »Hyperlinks« in Form von »Teleportern«, die dann wieder zur Verwirrung des Spielers beitragen, da zur Identifikation des Zielortes keine Information über den Weg dorthin herangezogen werden kann.

Abgesehen von den simulierten physikalischen Restriktionen hat der Spieler in der Navigation völlige Freiheit. Er kann sich nach Lust und Laune im Raum bewegen, d.h. Position und Blickrichtung wählen, solange er den virtuellen Boden unter den Füßen nicht verliert und auf kein Hindernis trifft. Diese Freiheit nützt dem Spieler allerdings nicht allzu viel, denn der schnellste Weg ist gefragt. Sie ist im Gegenteil sogar hinderlich, da es an Führung fehlt. Ein »ergonomischer« Spielprogramm ließe den Spieler immer nur in die jeweils richtige Richtung gehen, was freilich ziemlich langweilig wäre.

Die Gestaltung der Welt als Labyrinth stellt dem Spieler eine kognitive Aufgabe, welche kaum einer Erklärung bedarf. Zudem wird der Entdeckerdrang des Spielers bedient. Das inhärente Vordringen in unbekanntes Gebiet bildet einen idealen Rahmen für Überraschungen jeglicher Art, mithin für das chaotische Element. Gleichzeitig bietet es Raum für Dekoration und damit für eine Hinter-

grundgeschichte und die Schaffung einer Atmosphäre, die sich im Verlauf des Spiels auch ändern kann. Aufgabenangemessen ist ein Labyrinth bei nüchterner Betrachtung jedoch nicht. Es ist ja gerade so konstruiert, daß es den Spieler beim Finden und Zurücklegen des Weges nicht unterstützt, sondern behindert, etwa durch verschlossene Türen oder auch einfach dadurch, daß die Bewegung proportional zum zurückgelegten Weg Zeit kostet. Die Transparenz ist eingeschränkt; der Spieler weiß nicht, was vor ihm liegt und kann Objekte – Gegner, Waffen, den Ausgang, etc. – nur durch Besuch der entsprechenden Räume lokalisieren. Als Grundlage der Überraschung schafft das Labyrinth die Voraussetzungen für Verletzungen der Erwartungskonformität.

Mit den Gegnern wird das Konflikt-element ins Spiel eingebracht. Sie stellen sich dem Spieler in den Weg und bedrohen seinen Spielerfolg. Von Fehlertoleranz kann hier keine Rede sein; Fehler im Kampf mit Gegnern führen meist zum vorzeitigen Ende des Spiels oder einer langwierigen Suche nach Heilung und neuer Munition. Erschwerend kommt hinzu, daß sich die Spielfigur mit manchen Waffen selbst verletzen kann. Oft erscheinen Gegner plötzlich und unerwartet – Beldings *Chaos* läßt grüßen. Formal wird die Interaktion im Kampf noch vom Spieler gesteuert, doch eine Wahl des Tempos oder der Reihenfolge von Handlungen bleibt ihm kaum noch.

An anderen Stellen findet man hingegen Elemente guten User-Interface-Designs, nämlich dort, wo eine Erschwerung nicht zum Spielvergnügen, sondern nur zur Frustration des Spielers beitragen würde. So sind aufzufindende Gegenstände wie Schalter, Schlüssel und Waffen immer gleich gestaltet. Sie entsprechen mithin den Widgets im klassischen GUI, welche beispielsweise Buttons immer wie Buttons aussehen lassen. Auch in der Bedienung sind sie den Widgets recht ähnlich. Das Aufnehmen herumliegender Gegenstände etwa erfordert nicht etwa eine komplizierte, der Realität nachempfundene Abfolge von Bücken, Anfassen und Einpacken, sondern der Spieler bewegt seine Figur einfach zum Gegenstand und die Aufnahme erfolgt automatisch.

Die Grundlagen der Bedienung sind schnell gelernt. Perfekt gemeistert werden sie erst nach einer längeren Übungsphase, doch dieser Erwerb von Fertigkeiten ist Bestandteil des Spiels. Durch Variation der Anzahl, Bewaffnung oder Zähigkeit der Gegner läßt sich der Schwierigkeitsgrad des Spiels verändern, das Spiel also den Fähigkeiten seines Benutzers anpassen.

Die Herausforderung für den Spieler ergibt sich aus der Kombination der ungünstigen Faktoren und Schwierigkeiten, die innerhalb der Spielwelt jedoch plausibel erscheinen, zusammen mit Leistungsmessung zum Bei-

spiel durch Punktevergabe für gefundene Geheimräume und Zeitmessung. Dabei sind die zugrundeliegenden Regeln einfach und leicht erlernbar. Dreidimensionale Gestaltung und aus der physischen Welt übernommene Restriktionen unterstützen die Schaffung von Hindernissen. Funktionen, die solche Restriktionen aufheben, also etwa Wände durchsichtig machen oder ungehinderte Bewegung in jede Richtung erlauben, gelten als Mogelhilfe. Die Bewegungsfreiheit des Spielers im Raum macht das Spiel zum interessanten Zeitvertreib, unterstützt aber kaum beim Lösen der gestellten Aufgabe.

### 3.2. Adventures

#### 3.2.1. Beschreibung

Adventure-Spiele gehören nach Crawfords Klassifikation zu den Puzzles. Frühe Vertreter dieser Gruppe waren rein textbasiert und damit auf ausreichende Phantasie des Spielers angewiesen. Ihre modernen Nachfolger enthalten häufig auch Elemente interaktiver Geschichten, etwa in Form längerer Spielszenen. Klassiker des Genres sind beispielsweise die Reihen *Monkey Island* und *Space Quest*.

Auch im Adventure bewegt sich der Spieler in einer künstlichen Welt. Allerdings sieht er die gesteuerte Spielfigur meist als Zuschauer von außen. Echte Dreidimensionalität ist weniger wichtig; die Spielfigur bewegt sich von einem Bild zum nächsten und innerhalb der Bilder zwischen verschiedenen Punkten, aber die Bewegung selbst ist in der Regel bedeutungslos. Sie ist nichts weiter als das Ansteuern eines Punktes, im Prinzip ersetzbar durch einen einfachen Mausklick. Frühe Vertreter des Genres kamen sogar gänzlich ohne Grafik aus und stützten sich auf verbale Interaktion.

In der Spielwelt versteckt sind allerlei Gegenstände, die der Spieler untersuchen, manipulieren, zum Teil aufnehmen und miteinander kombinieren kann. Dazu müssen sie jedoch erst einmal aufgefunden werden, was zuallererst bedeutet, sie in der reichhaltig dekorierten Szene als bedeutsam auszumachen. Dies ist meist mit einem systematischen Absuchen des Bildschirms verbunden.

Neben Gegenständen kommen Personen (im weitesten Sinne, es kann sich je nach Hintergrundgeschichte auch um Fabelwesen o.ä. handeln) vor. Mit ihnen kann sich die Spielfigur unterhalten, wobei der Spieler den Dialog meist durch Auswahl aus einer Liste vorgegebener Sätze führt. Personen können aber auch auf von der Spielfigur herbeigeführte Ereignisse reagieren oder mit ihr Gegenstände austauschen.

Das Ziel des Spiels muß der Spieler oft erst im Verlauf der Handlung herausfinden. Es besteht typischerweise in

der Lösung eines finalen Puzzles. Als Sekundärziel müssen die nötigen Gegenstände oder Informationen gefunden werden, wozu wiederum Puzzles zu lösen sind. So ergibt sich eine Folge von Puzzles, die der Spieler nacheinander lösen muß. Nebenbei erkundet er die Spielwelt und durchquert sie oft (zumindest grob) in einer Richtung, von Szene zu Szene.

Adventure-Spiele können auch Action-Anteile enthalten, in denen die Fertigkeiten des Spielers in der Bedienung und seine Reaktionsfähigkeit getestet werden. Diese Teile haben jedoch geringe Bedeutung gegenüber den vielfältigen Puzzles. Die Bedienung kann meist vollständig mit der Maus erfolgen.

#### 3.2.2. Analyse

Auch hier soll sich die Betrachtung auf Konflikte, Chaos und Herausforderung konzentrieren. Konflikte kommen nur am Rande vor; es gibt keine Gegner, mit denen sich der Spieler über den gesamten Spielverlauf messen müßte. Gelegentlich stehen dem Spieler Personen im Wege, doch sie treten lediglich als Bestandteile eines zu lösenden Puzzles auf. Umso wichtiger ist, neben den motivierenden Elementen, das Überraschende und Unerwartete. Im Adventure manifestiert es sich vor allem in überraschenden Anwendungsmöglichkeiten von Gegenständen sowie in den Reaktionen des Spiels auf Lösung eines Puzzles. So muß der Spieler in *The Day of the Tentacle* zum Beispiel einen Wagen waschen, um Regen auszulösen – bekanntlich regnet es immer gerade dann, wenn man dies getan hat. Dank der verqueren Logik kommen dem Spieler die Zusammenhänge nicht so leicht in den Sinn. Hier liegt wieder ein Verstoß gegen das Prinzip der Erwartungskonformität vor. Manche Puzzles im Adventure basieren auf bekannten Zusammenhängen aus der realen Welt, andere verkehren Kausalbeziehungen ins Gegenteil. Eine ausführliche Betrachtung der verschiedenen Arten von Puzzles findet sich bei Bates [1].

Eine weitere Schwierigkeit und damit Herausforderung für den Spieler ergibt sich aus der Notwendigkeit, das Ziel des Spieles selbst zu ermitteln und im Spielverlauf Informationen zu sammeln, welche später nützlich sein könnten. Dazu muß das Spiel zweifellos selbstbeschreibend sein. Es ist dies jedoch in einer Art, die nicht ausschließlich auf Unterstützung des Spielers ausgerichtet ist. So können Informationen zum Beispiel lange vor dem Zeitpunkt ihrer Anwendung (und nur dort) vermittelt werden, also nicht unbedingt dort, wo der Spieler als Benutzer sie benötigt. Aufgabenangemessenheit und Puzzles widersprechen einander zumindest partiell. Die Aufgabe, ein Puzzle zu lösen, kann und soll vom Spiel angemessen un-

terstützt werden. Das Konzept des Puzzles selbst ist jedoch gerade auf hinreichende Schwierigkeit der Lösung gegründet.

Die Steuerbarkeit von Adventure-Spielen ist in der Regel gut; außerhalb von Action-Anteilen wird das Tempo vom Spieler bestimmt. Die Reihenfolge von Interaktionen wird jedoch durch die Puzzles vorgegeben, ist also systemseitig willkürlich festgelegt und muß vom Spieler erst ermittelt werden. Dafür sind solche Spiele in der Regel fehlertolerant. Der Spieler kann beliebig viele Versuche mit den verfügbaren Personen und Gegenständen anstellen und erhält anhand der Reaktionen oft sogar wertvolle Hinweise.

Die Navigation des Spielers erfolgt eher in einer diskreten Struktur denn in einem dreidimensionalen Raum, auch wenn als Darstellungsmittel 3D-Grafik verwendet wird. Insofern sind Adventures den klassischen Anwendungs-GUIs ähnlich. Die häufig zu findenden Vergrößerung eines Gegenstandes zwecks näherer Untersuchung oder Manipulation zum Beispiel weist Parallelen zu den Pop-up-Fenstern in solchen Anwendungen auf.

Schummeln kann der Spieler im Adventure nur durch Lesen einer Musterlösung. Klassische Adventures werden meist nur einmal gespielt. Jüngere Exemplare der Gattung erlauben jedoch verschiedene Handlungsstränge in Abhängigkeit von Entscheidungen des Spielers.

### 3.3. Simulations- und Strategiespiele

#### 3.3.1. Beschreibung

Unter Simulations- und Strategiespielen, nach Crawford teils als *Spielzeuge (Toys)*, teils als *Spiele (Games)* zu klassifizieren, sollen hier Spiele wie *SimCity*, *Civilization* und *Die Siedler* verstanden werden. Flug- oder Sportsimulation und ähnliche Programme werden ausgeklammert.

Hier sieht der Spieler eine Welt in der Draufsicht. Dreidimensionale Grafik wird bei der Darstellung darauf befindlicher Objekte benutzt. Im wesentlichen findet das Spiel jedoch, ähnlich einem Brettspiel, auf einer zweidimensionalen Fläche statt, die in Felder eingeteilt ist. Auf diesen Feldern können zum Beispiel Gebäude oder Städte errichtet werden, sie repräsentieren Landschaftsformen oder Ressourcen, oder militärische Einheiten lassen sich von einem Feld in ein benachbartes bewegen.

Aufgabe des Spielers ist der sukzessive Aufbau einer virtuellen Welt, deren Gedeihen von komplexen, nicht immer offensichtlichen Zusammenhängen abhängt. Das Spiel ist dabei in eine Vielzahl von Einzelaktionen unterteilt, deren Gesamtwirkung schließlich zum Gewinn führt, im Spiel *Civilization* zum Beispiel zu einem Staat,

der in der Lage ist, als erster fremde Planeten zu besiegen. Dabei kämpft der Spieler entweder nur gegen die der Spielwelt innwohnenden Widrigkeiten, oder aber gegen andere Spieler, real oder vom Computer simuliert, die dasselbe Ziel in derselben Welt verfolgen. Neben Interaktionen mit der Weltdarstellung kommen häufig klassische GUI-Prinzipien wie Menüs und Dialoge zum Einsatz, mit denen der Spieler zum Beispiel Steuersätze festlegen kann. Bereits Aufgebautes kann durch unerwartete Ereignisse – simulierte Naturkatastrophen, Kriege o.ä. – zerstört werden.

Eine Bewegung des Spielers durch die Welt erfolgt in der Regel nicht. Statt dessen erhält er alle nötigen Informationen aus der Übersichtsdarstellung und Parameteranzeigen und handelt in Form von Befehlen an Objekte oder Subjekte auf dem Spielfeld.

#### 3.3.2. Analyse

Hier steht der Spieler außerhalb des Spielgeschehens, er ist gewissermaßen Gott. Physische Eigenschaften und bildliche Darstellungen spielen eine Untergeordnete Rolle. Simulationsspiele sind abstrakter; die Darstellung ähnelt nicht zufällig der eines modernen Brettspiels. Der Spieler manipuliert Objekte und Parameter, sonst nichts. Das überraschende, chaotische Moment findet sich auch hier wieder, jedoch spielt es, sind die Regeln erst einmal ergründet, eine weniger bedeutende Rolle. Fast alle Kriterien der Softwareergonomie werden erfüllt. Lediglich die grafischen Mittel der Darstellung, das äußere Design, unterscheidet das User Interface eines solchen Spiels von gängigen GUIs. Das bringt Spaß und wegen der speziellen Funktionen und Aufgaben der Objekte im Spiel sind ohnehin eigens gestaltete »Widgets« nötig; völlige Konsistenz etwa mit dem Look-and-Feel von Windows wäre eher hinderlich.

Navigation im engeren Sinne findet sich folgerichtig nicht. Statt dessen werden Spielzüge innerhalb eines festen Rahmens von Möglichkeiten ausgeführt.

Die Herausforderung und Grundlage der Faszination dieser Simulationen liegt – neben dem Kampf mit den Gegnern – in den verdeckten Parametern. Der Spieler beeinflußt sie unweigerlich mit jeder Interaktion und jeder Entscheidung, aber er kann die Parameter und die Zusammenhänge zwischen ihnen nicht völlig überblicken. Die Transparenz und Selbstbeschreibung ist also stark eingeschränkt.

## 4. Schlußfolgerungen

### 4.1. Einige Hypothesen

Die vorstehenden Betrachtungen führen zu einigen Hypothesen, welche vor einer gedankenlosen Übertragung von Konzepten aus der Welt der Computerspiele in jene der »ernsthaften« Anwendungen zu überprüfen sind.

#### 4.1.1. Computerspiele sind nicht einfach

Um ein über längere Zeit fesselndes Spielerlebnis zu schaffen, müssen Computerspiele gegen Grundregeln der Softwareergonomie verstößen. Das ideale Anwendungsprogramm ist gewissermaßen unsichtbar. Es erlaubt seinem Benutzer volle Konzentration auf seine Aufgabe und unterstützt immer dann und dort, wo Hilfe nützlich ist. Spiele hingegen müssen zum einen erst künstlich Aufgaben schaffen, die der Spieler ohne das Spiel gar nicht zu lösen hätte, und zum anderen die Lösung der Aufgaben ausreichend erschweren, um Langeweile und Desinteresse zu vermeiden. Anwendungen sollen unterstützen, Spiele herausfordern. Zudem dienen Spiele dem Zeitvertreib, während Anwendungsprogramme die effiziente Erledigung von Aufgaben unterstützen müssen.

Die verschiedenen Klassen von Computerspielen bedienen sich unterschiedlicher Mittel, um künstlich eine Herausforderung zu schaffen. Simulations- und Strategiespiele erfüllen formal die meisten Regeln guter User-Interface-Gestaltung. Die Schwierigkeiten und Herausforderungen schafft das zugrundeliegende Modell zusammen mit der unvollständigen Informationsdarstellung. Der Spieler sieht Auswirkungen, aber nicht unbedingt Ursachen und Zusammenhänge. Bei den Adventures verhält es sich ähnlich. Die grafische Darstellung erzählt eine Geschichte rund um die Puzzles und erlaubt deren nonverbale Darstellung, während die zu lösenden Aufgaben vergleichsweise abstrakt und unabhängig von der gewählten Darstellungsform sind. 3D-Shooter hingegen lassen keine deutliche Grenze zwischen ihrem grafischen User Interface und einem zugrundeliegenden Modell erkennen. Bedienung und abstrakte Spielhandlung verschmelzen zu einem Ganzen, die Bedienung selbst ist das Spiel.

#### 4.1.2. Ergonomie kann auf verschiedenen Ebenen verletzt werden

Computerspiele sind schwer. Gleichzeitig müssen sie einfach sein, sonst würde sie niemand spielen. Ein Widerspruch? Nein. Computerspiele verstößen dort gegen Ergonomieregeln, wo dies im Rahmen der Spielhandlung

plausibel und notwendig ist. Die grundlegende Interaktion hingegen wird genau wie in Anwendungsprogrammen so einfach wie möglich gemacht. In einem 3D-Shooter wird der Spieler nicht daran scheitern, daß er einen gefundenen Gegenstand nicht identifizieren oder benutzen kann.

Der Spieler nimmt Verletzungen der Ergonomie nicht als solche wahr, da sie nicht als Belästigung, als ärgerliches Hindernis erscheinen, sondern als eine Herausforderung, die es zu meistern gilt. Gute Computerspiele sind nicht schlechthin einfach, sondern sie halten die Einstiegsschwelle niedrig und setzen Erschwernisse in einer wenig frustrierenden Weise ein. Hierin liegt die Kunst des Spieldesigns.

#### 4.1.3. Virtuelle Umgebungen sind unpassend für die meisten Anwendungen

Selbst ein Spielerlebnis läßt sich auch mit eher abstrakten Darstellungen vermitteln. Der Autor beispielsweise hat mit dem Spiel *Civilization* und dessen zweidimensionaler Darstellung wesentlich mehr Zeit verbracht als mit allen 3D-Shootern zusammen. Ein vielleicht noch besseres, weil noch abstrakteres Beispiel bietet das Schachspiel. Die Übertragung in eine virtuelle Umgebung liefert hier keinerlei Mehrwert; die im wesentlichen zweidimensionale Übersichtsdarstellung ist dem Spiel angemessen. Gleiches gilt vermutlich für viele Arten von Anwendungsprogrammen. Niemand käme wohl auf die Idee, eine virtuelle Schreibmaschine in einem Holodeck als Textverarbeitungswerkzeug der Zukunft vorzuschlagen.

Räumliche Navigation in einer 3D-Welt kann zuviel Freiheitsgrade für eine konkrete Anwendung bieten, selbst wenn dreidimensionale Darstellungen sinnvoll sind. In einem Adventure-Spiel zum Beispiel würde die Möglichkeit einer völlig freien Wahl von Standort und Blickrichtung von den Puzzles, dem eigentlichen Spielinhalt also, ablenken und zudem den Raum vergrößern, der nach Gegenständen und Hinweisen abgesucht werden muß. Werden die zu handhabenden Informationen und Probleme abstrakter, so muß auch die Darstellung abstrakter werden und die Navigation eher in den Informationsstrukturen als in einem davon unabhängigen System von Freiheitsgraden erfolgen.

#### 4.1.4. Realistische 3D-Welten sind nicht benutzerfreundlich

Das Beispiel der 3D-Shooter zeigt, daß realistische 3D-Welten mit aus der realen Welt übertragenen Eigenschaften zur Erschwerung der Interaktion beitragen können.

Das Herumlaufen kostet unnötig Zeit, Objekte verschwinden, sobald die Sichtbeziehung abbricht, und die Anordnung von Objekten folgt nicht logischen, aufgabenorientierten Kriterien, sondern physischen Restriktionen, welche in einem Computerprogramm selten einen Sinn haben. Es sei denn, man möchte damit Zeit totschlagen. Eine ausführliche Betrachtung zum Realismus im Interface-Design findet sich in der Interface Hall of Shame [?].

Selbst die vordergründig sehr realistischen 3D-Shooter verzichten auf Realismus, wo er nicht nahtlos in die Spielhandlung eingebettet werden kann und den Spieler eher belästigen als anregen würde. Möglicherweise spielen dabei auch Beschränkungen der Eingabegeräte eine Rolle. Das Aufhaben eines Gegenstandes zum Beispiel lässt sich mit Maus und Tastatur nur mit einem gewissen Abstraktionsgrad darstellen, wenn die Bedienung nicht zu umständlich und erklärmungsbedürftig sein soll.

Wesentliche Elemente realistischer Darstellung sind physikalische Restriktionen einerseits und die völlige Freiheit innerhalb der dadurch gesetzten Grenzen andererseits. Jedes für sich kann dem Benutzer einer Anwendung Schwierigkeiten bereiten. Spiele können diese Schwierigkeiten nutzen, da sie den Spieler beschäftigen und aus sich heraus einen Anwendungszweck generieren müssen. Anwendungsprogramme hingegen dienen einem äußeren Zweck, dem sie sich völlig unterordnen müssen.

## 4.2. Weitere Forschung

Das effektivste Mittel zur Untersuchung der Softwareergonomie sind Benutzertests [11, 13]. Neben der Prüfung der hier aufgestellten Hypothesen scheint ein kritischer Vergleich von VE-Systemen mit klassischen GUI-Lösungen des jeweils gleichen Einsatzzwecks angebracht. Solche Tests sind bisher oft daran gescheitert, daß VE-Systeme in der Vergangenheit oft für völlig neuartige Anwendungen vorgeschlagen wurden. Dies kann jedoch kein Argument dafür sein, Alternativen, selbst wenn sie weniger innovativ erscheinen mögen, nicht zu untersuchen. Nützlich könnte ein allgemeiner Kriterienkatalog sein, nach dem sich Probleme und Anwendungsbereiche hinsichtlich der Eignung von VR-Techniken untersuchen lassen. Mögliche Kriterien wären zum Beispiel die Frage nach der Adequatheit räumlicher Navigation und, eng damit verbunden, die nach dem angemessenen Abstraktionsgrad.

## Literatur

- [1] B. Bates: „Designing the Puzzle“, *Proceedings of the 1997 Game Developers Conference*, <http://www.scottkim.com/thinkinggames/GDC00/bates.html>
- [2] T. Belding: *Elements of Videogame Style*, <http://hamilton.htcomp.net/tbelding/oddities/elements.txt>
- [3] *Bildschirmarbeitsverordnung*, <http://www.bma.de/download/gesetze/bildschirmarbeit.htm>
- [4] C. Crawford: *The Art of Computer Game Design*, <http://www.vancouver.wsu.edu/fac/peabody/game-book/Coverpage.html>
- [5] C. Crawford: „My Definition of »Game«“, *The Journal of Computer Game Design*, Vol. 4, [http://www.erasmatazz.com/library/JCGD\\_Volume\\_4/DefinitionGame.html](http://www.erasmatazz.com/library/JCGD_Volume_4/DefinitionGame.html)
- [6] C. Crawford: „Design Review and Analysis: Doom“, *Journal of Computer Game Design*, Vol. 7, [http://www.erasmatazz.com/library/JCGD\\_Volume\\_7/Doom\\_Review.html](http://www.erasmatazz.com/library/JCGD_Volume_7/Doom_Review.html).
- [7] C. Drolshagen: *Duke Nukem 3D. Die Verwandlung eines 3D-Shooters in ein Kooperationsspiel*. Fachhochschule Köln, Diplomarbeit, <http://www.drolshagen.net/diplom/Diplomarbeit.html>
- [8] E. Eberleh u.a.: *Einführung in die Software-Ergonomie*, Berlin, New York: de Gruyter, 1994.
- [9] C. Faulkner: *The Essence of Human-Computer Interaction*, Prentice Hall, 1998.
- [10] J. Nielsen: *Usability Engineering*, Boston: AP Professional, 1993.
- [11] J. Nielsen und R.L. Mack (Hrsg.): *Usability Inspection Methods*, New York: John Wiley & Sons, 1994
- [12] D. A. Norman: *The Psychology of Everyday Things*, New York: Basic Books, 1988.
- [13] D. A. Norman: „Affordance, conventions, and design“, in: *Interactions* 6, 3 (May 1999).
- [14] J. Raskin: *The Humane Interface*, Addison Wesley Longman, 2000.
- [15] B. Shneiderman: *Designing the User Interface*, 2nd ed. Addison-Wesley, 1992.