

# Dynamisches Objektverhalten in virtuellen Umgebungen



Vom Fachbereich Informatik  
der Technischen Universität Darmstadt  
genehmigte

## Dissertation

zur Erlangung des akademischen Grades eines  
Doktor-Ingenieurs (Dr.-Ing.)  
von

Dipl. Inform. Torsten Fröhlich  
aus Bonn

Referenten der Arbeit: Prof. Dr. h.c. Dr.-Ing. José L. Encarnação  
Prof. Dr. Stefan Müller

Datum der Einreichung: 19. August 2002  
Datum der mündlichen Prüfung: 4. Oktober 2002

D17  
Darmstädter Dissertationen 2002



# Danksagung

Eine Reihe von Menschen hat einen Anteil an dieser Arbeit, direkt durch intellektuelle Anregung oder Beiträge und indirekt durch Unterstützung und Geduld. Bei diesen Menschen möchte ich mich herzlich bedanken.

Mein erster Gedanke gilt meiner Familie. Ihre Ermutigung den von mir gewählten Lebensweg zu gehen, verbunden mit Verständnis und Anteilnahme wenn es scheinbar nicht voran geht, war mir Zeit meines Lebens ein großer Rückhalt.

Dafür, daß ich in den letzten sieben Jahren in einem einmalig ausgestatteten, hoch motiviertem Team an interessanten, teilweise geradezu spannenden Aufgaben arbeiten durfte, möchte ich mich bei Prof. José L. Encarnação und bei Prof. Stefan Müller bedanken. Das interessante Forschungsgebiet und die Ermunterung beider Referenten hat mich immer wieder motiviert, die Promotion nicht aus den Augen zu verlieren.

Meinen Kollegen im Haus der Grafischen Datenverarbeitung, ganz besonders meinen Kollegen in der Abteilung "Virtuelle und Erweiterte Realität" möchte ich für die immer kameradschaftliche, manchmal freundschaftliche Zusammenarbeit danken. Teil dieses Teams gewesen zu sein, empfinde ich als Auszeichnung.

Bedanken möchte ich mich auch bei den vielen Studenten, die ich in den letzten Jahren als Diplomanden und wissenschaftliche Hilfskräfte betreut habe. Mit Fleiß und Zuverlässigkeit haben sie mich in meiner Projektarbeit unterstützt. Ihre Beiträge und Anregungen finden sich auch in der vorliegenden Arbeit wieder.

Mein besonderer Dank gilt Marcus Roth, zunächst Diplomand, später geschätzter Kollege am Fraunhofer-IGD. Seine Arbeit und der wissenschaftliche Austausch mit ihm waren für mich außerordentlich wertvoll und haben in der vorliegenden Dissertationsschrift an vielen Stellen Niederschlag gefunden.



# Inhaltsverzeichnis

<b>1</b>	<b>Einleitung</b>	<b>5</b>
1.1	Motivation und Einführung . . . . .	5
1.1.1	Virtuelle Realität: Entwicklung und Anwendungen . . .	5
1.1.2	Dynamisches Objektverhalten . . . . .	8
1.2	Aufgabenstellung und Zielsetzung . . . . .	10
1.3	Zusammenfassung der wichtigsten Ergebnisse . . . . .	11
1.4	Gliederung der Arbeit . . . . .	12
<b>2</b>	<b>Stand der Wissenschaft und Technik</b>	<b>13</b>
2.1	VR-Systeme . . . . .	14
2.1.1	Simple Virtual Environment Library . . . . .	14
2.1.2	Minimal Reality . . . . .	16
2.1.3	Alice . . . . .	17
2.1.4	Sense8 World Tool Kit . . . . .	19
2.1.5	VRML-basierte Systeme . . . . .	21
2.2	Systeme zur Anbindung von Interaktionsgeräten . . . . .	22
2.2.1	Microsoft DirectInput . . . . .	22
2.2.2	Fraunhofer-IGD INTO . . . . .	24
2.3	Vergleich und Bewertung . . . . .	25
2.3.1	Systemarchitektur . . . . .	25
2.3.2	Struktur der Datenbasis . . . . .	26
2.3.3	Parallelisierung von Darstellung und Simulation . . . .	29
2.3.4	Objektmodell . . . . .	29
2.3.5	Statisches und dynamisches Verhalten . . . . .	30
2.3.6	Integration von multidimensionalen Interaktionsgeräten	31
2.4	Zusammenfassung . . . . .	32
<b>3</b>	<b>Ein Simulationssystem für Autonome Objekte</b>	<b>34</b>
3.1	Einleitung . . . . .	34
3.2	Entwurfsprozeß . . . . .	35
3.3	Trennung von grafischer Darstellung und Simulation . . . . .	37

3.4	Autonome Objekte . . . . .	39
3.5	Kommunikation über Nachrichten . . . . .	41
3.6	Das Simulationssystem . . . . .	41
3.6.1	Die Simulationsschleife . . . . .	41
3.6.2	Struktur Autonomer Objekte . . . . .	42
3.6.3	Nachrichten Kommunikation . . . . .	43
3.7	Features . . . . .	44
3.7.1	Attribute . . . . .	44
3.7.2	Operationen auf Attributen . . . . .	45
3.8	Protokolle . . . . .	45
3.8.1	Konfiguration Autonomer Objekte zur Laufzeit . . . . .	46
3.8.2	Adressierung von Autonomen Objekten . . . . .	47
3.9	Fehlerbehandlung . . . . .	48
3.9.1	Fehler im Programmcode . . . . .	49
3.9.2	Protokollfehler . . . . .	49
3.9.3	Zuordnungsfehler . . . . .	50
3.9.4	Quittungen und Fehlerquittungen (NACK) . . . . .	50
3.9.5	Architektur des Simulationssystems . . . . .	51
3.10	Prozeßstruktur . . . . .	52
3.11	Parallelisierung der Simulation . . . . .	53
3.12	Die Welt-Datenbasis . . . . .	54
3.13	Einteilung der Welt-Datenbasis in ein Gitter . . . . .	56
3.13.1	Verfahren zur Erkennung von Hindernissen . . . . .	56
3.13.2	Verfahren zur Pfadsuche in Virtuellen Umgebungen . . . . .	57
<b>4</b>	<b>Modellierung von Simulationen mit Autonomen Objekten</b>	<b>63</b>
4.1	Einleitung . . . . .	63
4.2	Eine Notation zur Darstellung von Feature-Klassen . . . . .	65
4.2.1	Abhängigkeit, Signatur und Attribute . . . . .	65
4.2.2	Protokoll . . . . .	66
4.2.3	Pseudocode . . . . .	66
4.3	Grundlegende Konzepte . . . . .	67
4.3.1	Kontrolle . . . . .	67
4.3.2	Verteilung von Daten: Abonnieren . . . . .	68
4.3.3	Grafische Objekte . . . . .	69
4.3.4	Kollisionserkennung . . . . .	70
4.4	Benutzerinteraktion . . . . .	72
4.4.1	Anbindung der Interaktionsgeräte . . . . .	72
4.4.2	Cursor . . . . .	72
4.4.3	Greifen . . . . .	73
4.5	Benutzer Modell . . . . .	74

<b>5</b>	<b>Integration multidimensionaler Interaktionsgeräte</b>	<b>78</b>
5.1	Einleitung . . . . .	78
5.2	Interaktionsgeräte . . . . .	79
5.2.1	Eingabegeräte . . . . .	79
5.2.2	Ausgabegeräte . . . . .	82
5.3	Aspekte der Geräteintegration und abgeleitete Forderungen . .	84
5.4	Das IDEAL System . . . . .	87
5.4.1	Systemarchitektur . . . . .	87
5.4.2	Demon Prozesse . . . . .	90
5.4.3	Server Prozesse . . . . .	91
5.4.4	Die logische Geräteschnittstelle von IDEAL . . . . .	92
5.4.5	Zugriff auf logische Geräte . . . . .	95
5.4.6	Kommunikation mit zwischen Geräteschnittstelle und Servern . . . . .	97
5.4.7	Transformation von 6D-Gerätedaten . . . . .	98
5.4.8	Latenz und deren Reduktion durch optimierte Kom- munikationsprotokolle . . . . .	99
5.4.9	Die grafische Benutzungsoberfläche . . . . .	104
5.5	Zusammenfassung . . . . .	109
<b>6</b>	<b>Demonstration von Anwendungen</b>	<b>110</b>
6.1	Astronauten Trainingssimulator . . . . .	110
6.1.1	Trainingsszenario . . . . .	110
6.1.2	Technische Installation . . . . .	111
6.1.3	Einsatz Autonomer Objekte . . . . .	112
6.1.4	Nachrichten zum Greifen der Aggregate . . . . .	113
6.1.5	Synchronisation der Szenengraphen . . . . .	114
6.1.6	Das Experiment . . . . .	114
6.2	Das Virtuelle Ozeanarium . . . . .	115
6.2.1	Einleitung . . . . .	115
6.2.2	Verwandte Arbeiten . . . . .	116
6.2.3	Systemarchitektur des Virtuellen Ozeanariums . . . . .	117
6.2.4	Aufbau und Inhalte des Virtuellen Ozeanariums . . . . .	118
6.2.5	Modellstruktur . . . . .	120
6.2.6	Ablaufsteuerung . . . . .	121
6.2.7	Simulation von Lichteffekten unter Wasser . . . . .	123
6.2.8	Virtuelle Kreaturen . . . . .	125
6.2.9	Simulationsmodell virtueller Fische . . . . .	125
6.2.10	Parallelisierung der Simulation . . . . .	136
6.2.11	LOD in der Simulation . . . . .	136
6.2.12	Quantitative Simulation . . . . .	140

<i>INHALTSVERZEICHNIS</i>	4
6.2.13 Integration von visueller und quantitativer Simulation .	142
6.3 Der Dom von Siena . . . . .	145
6.3.1 Installation . . . . .	145
6.3.2 Autonome Objekte für virtuelle Charaktere . . . . .	146
<b>7 Zusammenfassung und Ausblick</b>	<b>150</b>

# Kapitel 1

## Einleitung

### 1.1 Motivation und Einführung

#### 1.1.1 Virtuelle Realität: Entwicklung und Anwendungen

Der Begriff Virtuelle Realität (VR) bezeichnet eine Technologie, welche die Illusion erzeugt, sich in einer computergenerierten dreidimensionalen Welt zu befinden und in dieser zu interagieren. Die Illusion in eine virtuelle Welt einzutauchen bezeichnet man als *Immersion*. Die computergenerierten Welten selbst werden als Virtuelle Welten oder Virtuelle Umgebungen bezeichnet.

Immersion entsteht, indem der Benutzer über sogenannte multidimensionale Interaktionsgeräte wie Trackingsysteme und Datenhandschuhe mit der Virtuellen Umgebung interagiert. Werden diese am Körper getragen oder an der Kleidung befestigt, nimmt der Benutzer sie kaum noch wahr. Die Interaktion erfolgt intuitiv über Körperbewegungen, Gesten und Sprache. Das System reagiert in Echtzeit auf die Eingaben und passt die audiovisuelle Darstellung augenblicklich an. Stereodarstellung von Bild und Ton fügen räumliche Tiefe hinzu und verstärken so die Immersion.

Neben dem Sehsinn und dem Gehör wird in einigen Fällen auch der Tastsinn mit einbezogen. Erste Geräte, sogenannte haptische Displays, existieren bereits [Zieg98]. Geruchs- und Geschmackssinn dagegen werden bisher kaum angesprochen.

Zur Erzeugung der Immersion sind leistungsfähige Grafikkrechner erforderlich, die die dreidimensionale Darstellung einer Szene augenblicklich, d.h. ohne wahrnehmbare Verzögerung, den Aktionen des Benutzers folgend aktualisieren. Solche Rechner stehen seit Ende der 80er Jahren zur Verfügung und seitdem hat ihre Leistungsfähigkeit enorm zugenommen.

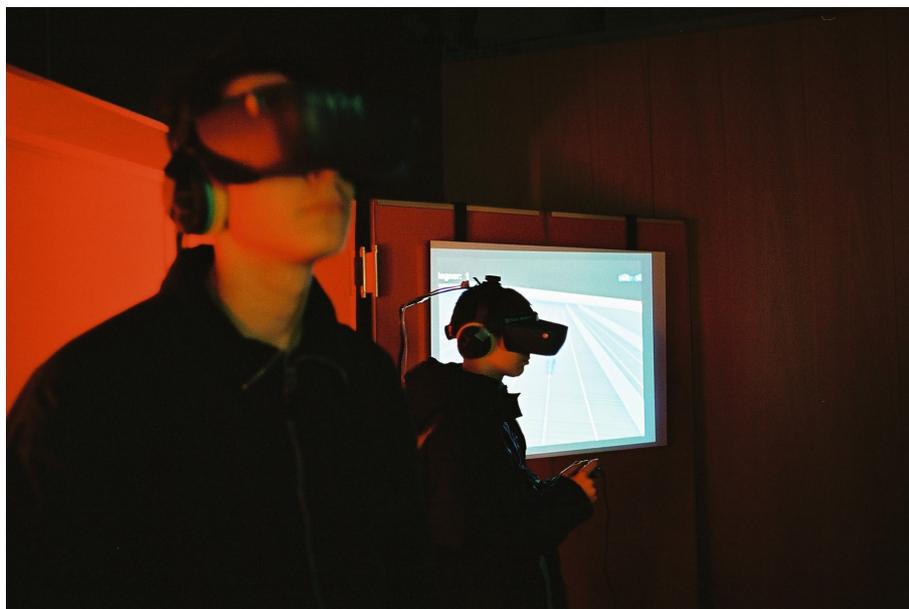


Abbildung 1.1: Interaktion in einer Virtuellen Welt

1990-1995 war das Pionierzeitalter für VR (erste VR-Generation), wobei verschiedene Forschungsgruppen erste Beispiele entwickelten und fieberhaft nach Anwendungen suchten [BBHL90, Brys94]. Diese waren ca. 1995 speziell im Maschinenbau gefunden [AFFG96] und wurden in der zweiten VR-Generation (1995-2000) weiter ausgefeilt. So werden heute im Automobil- und Flugzeugbau VR-Methoden im Bereich der "digitalen Produktentwicklung" flächendeckend eingesetzt. Ziel ist es, Produkte schneller und kostengünstiger zu entwickeln, und diese schneller auf den Markt zu bringen. Anstatt reale Modelle und Prototypen zu bauen, werden dabei verstärkt digitale VR-Modelle eingesetzt und begutachtet. Diese zweite VR-Generation war aber immer noch gekennzeichnet durch hohe Investitionskosten (Hardwarekosten in Millionenhöhe). Damit war der Zugang zu VR-Technologien nur Forschungseinrichtungen und der Großindustrie vorbehalten[Muel02].

Angetrieben durch die schnell wachsende Computerspiele-Industrie hat sich in den letzten Jahren besonders die Grafikleistung der Personal Computer (PC) rasant entwickelt. Moderne PC sind mit 3D-Grafikkarten ausgestattet, deren Leistung inzwischen die der klassischen Grafikworkstations teilweise weit übersteigt, deren Anschaffungskosten aber nur einen Bruchteil betragen. Auch die Kosten für Projektionssysteme sind um Größenordnungen gesunken; Projektoren sind heute für wenige tausend Euro zu haben.

Damit wird Virtuelle Realität plötzlich für öffentliche Einrichtungen, die

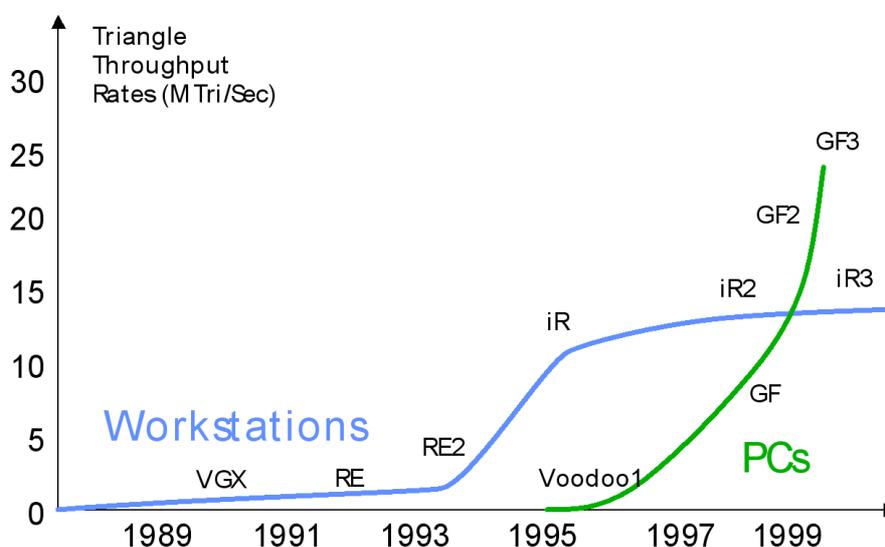


Abbildung 1.2: Entwicklung der Grafikleistung

mittelständische Wirtschaft und letztlich auch Privatanwender verfügbar. Für eine Vielzahl von Anwendungen der Virtuellen Realität entstehen völlig neue Märkte. Dies gilt besonders für die Bereiche Training [Bowe96, FDFM96, LYDS01], öffentliche und berufliche Bildung und Unterhaltung [AWHW96, Froh97, LuWe99, LuFr00, Froh00]. Zwar konnte in den letzten Jahren ein großes Potential für diese neuen Technologien aufgezeigt werden [BEFK02], die entwickelten Systeme waren für die Öffentlichkeit aber überhaupt nicht oder nur zeitlich begrenzt, etwa auf Ausstellungen oder Messen, zugänglich.

Besonders im Bereich Edutainment<sup>1</sup> besitzt die Virtuelle Realität enorme Möglichkeiten. Sie erlaubt Reisen an unzugängliche Orte, in den Mikro- oder Makrokosmos etwa, oder in die Vergangenheit oder Zukunft. Die Möglichkeit, in Rollen zu schlüpfen, etwa als Taucher die Unterwasserwelt des Ozeans selbst zu entdecken, eröffnet ganz neue pädagogische Perspektiven. Durch das Erleben von Virtuellen Welten kann mit Spaß nachhaltig gelernt werden. Die Faszination der Technik bietet ein Transportmittel für Wissen auch an Personen, die mit den eigentlichen Inhalten vielleicht nicht angesprochen worden wären. Daß VR und ihre jüngere Schwester Augmented Reality enorm attraktiv für die Öffentlichkeit sind, wird in [Froh00, BFKK01] und [BEFK02] eindrucksvoll bewiesen.

Damit dieses didaktische Potential ausgeschöpft werden kann, müssen Virtuelle Welten jedoch wesentlich lebendiger werden, als es für den Einsatz

<sup>1</sup>Kunstwort zusammengesetzt aus Entertainment (Unterhaltung) und Education (Bildung, Ausbildung)

in der Industrie nötig gewesen ist. Geeignete VR Systeme müssen einfacher zu bedienen sein und mehr echte Interaktion zulassen.

Grundlage für diese VR-Systeme der dritten Generation sind die einfache Handhabung und Anbindung von Interaktionsgeräten sowie dynamisches Objektverhalten, welches lebendig wirkende Virtuelle Umgebungen ermöglicht, deren Inhalt über starre, vorgegebene Abläufe hinausgeht. Diesen Themen widmet sich die vorliegende Arbeit.

### 1.1.2 Dynamisches Objektverhalten

Von *Verhalten* in Virtuellen Umgebungen wird gesprochen, wenn die Objekte, aus denen die Umgebung besteht, ihre Eigenschaften verändern können. Der Benutzer nimmt die Veränderung von Lage, Form, Oberfläche, Klang usw. wahr. VR-Systeme der ersten und zweiten Generation weisen fast ausschließlich *statisches* Verhalten auf, welches bei der Implementierung einer Anwendung festgelegt wird und sich zu deren Laufzeit nicht mehr ändert. *Dynamisches* Objektverhalten dagegen erlaubt dem Benutzer zur Laufzeit das Verhalten tatsächlich zu beeinflussen. So können lebendige Virtuelle Welten entstehen, die wirkliche Interaktion erst ermöglichen. Systeme der dritten Generation sollten solches dynamisches Verhalten ermöglichen. Abbildung 1.3 bildet das Kontinuum von dreidimensionalen Grafikanwendungen ab und ordnet die Generationen von VR-Systemen neben anderen Nicht-VR Anwendungen ein.

Zu Beginn der 90er Jahre waren sogenannte Walkthroughs häufig anzutreffen, in denen die einzige Form der Interaktion in der Navigation bestand, also der Möglichkeit, seinen Standort und die Blickrichtung zu verändern. Die umgebende Szene war völlig statisch. Walkthroughs finden bis heute in der Architekturvisualisierung Anwendung.

VR-Systeme der ersten Generation erlaubten Basisinteraktion und statisches Objektverhalten. Der Benutzer konnte Objekte greifen, bewegen und manipulieren. Animationen stellten dynamische Vorgänge dar. Ergebnisse von Simulationen wurden in virtuellen Umgebungen visualisiert [Brys94]. Das Verhalten wurde zunächst über Programmcode (im VR-System selbst oder in anwendungsspezifischen Modulen enthalten) definiert. Anwendungen bestanden in Einbau-Montage Simulationen und dem Design Review [AFFG96].

Die zweite Generation von VR-Systemen, erlaubte es zunehmend Objektverhalten auch über Skriptsprachen oder das "verdrahten" von Objekten untereinander flexibler und einfacher zu definieren [Conw00, WUP00]. Die Beschreibung des Verhaltens und der Beziehungen zwischen den Objekten einer Szene und zum Benutzer wird bis heute bei der Entwicklung der Virtuellen Umgebung festgelegt und kann sich zur Laufzeit nicht mehr

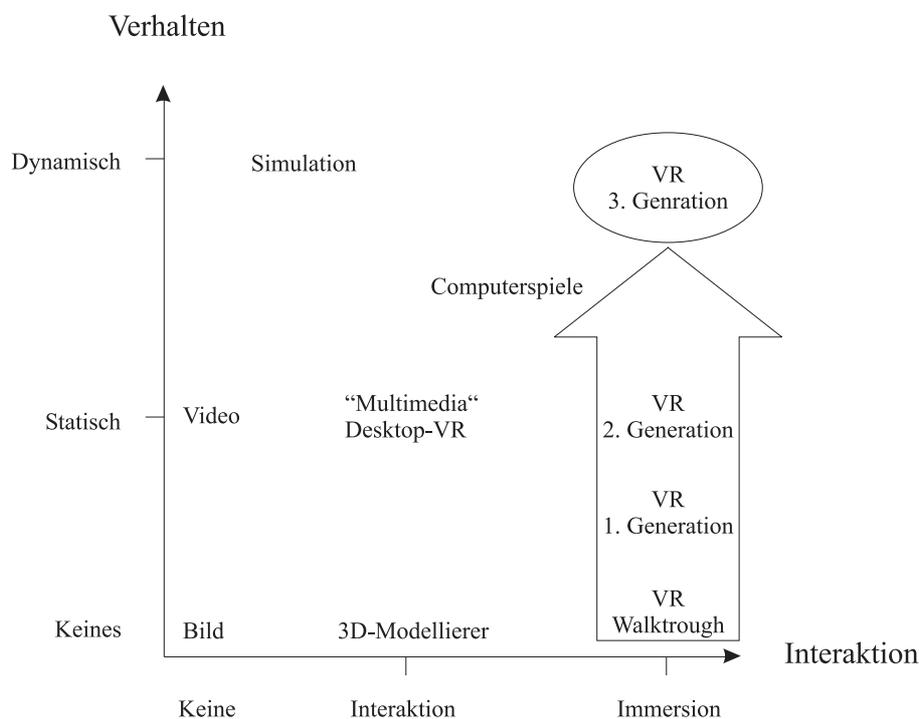


Abbildung 1.3: Kontinuum der 3D Grafikanwendungen

ändern, bleibt also statisch. Daher sprechen wir in diesem Zusammenhang von *statischem* Verhalten. Tatsächlich sehen wir heute VR-Anwendungen, in denen nur einige wenige Objekte manipuliert werden können und Dynamik durch festgelegte Animationen nur vorgetäuscht wird. Die Möglichkeiten zur Einflussnahme bestehen hauptsächlich aus dem auslösen von vorgegebenen Abläufen und der Navigation.

VR-Systeme der dritten Generation werden komplexe, lebendige Welten ermöglichen und dem Benutzer erlauben, auf das Verhalten seiner Umgebung Einfluss zu nehmen. Objekte innerhalb der Szene werden untereinander interagieren und sich so gegenseitig in ihrem Verhalten beeinflussen. Die Dynamik der Virtuellen Umgebung wird durch Simulation und nicht durch vorgerechnete Animationen dargestellt. Dies wird mit dem Begriff *dynamisches Verhalten* ausgedrückt.

Dynamisches Verhalten ist aber mit den gegenwärtig verfügbaren Systemen nur schwer zu realisieren. Virtuelle Umgebungen mit einer großen Zahl beweglicher Objekte, die dynamisches Verhalten aufweisen, wie dies beispielsweise für Simulationen von Verkehr oder künstlichem Leben in erforderlich ist, findet man bisher konsequenterweise vergleichsweise selten [TuTe94, Reyn87] und es handelt sich dabei um Speziallösungen die mit nicht uner-

heblichem Entwicklungsaufwand realisiert werden mussten.

Für die Entwicklung von Virtuellen Umgebungen mit dynamischem Verhalten müssen immer wieder die gleichen Probleme gelöst werden, z.B. die parallele Abarbeitung von Simulationsaufgaben, die Suche nach Objekten mit bestimmten semantischen Eigenschaften oder in einem Volumen im dreidimensionalen Raum und die Kommunikation von Ereignissen innerhalb der Virtuellen Umgebung. Die gegenwärtig existierenden VR-Systeme und Autorenwerkzeuge unterstützen den Entwickler bei der Lösung solcher Aufgaben nur ungenügend.

Die Entwicklung von Systemen mit dynamischem Objektverhalten besser zu unterstützen ist ein Anliegen dieser Arbeit.

Ein weiterer Schwerpunkt der Arbeit befasst sich mit den Interaktionsgeräten, die in der Virtuellen Realität Verwendung finden. Ihre Benutzung erlaubt zwar die intuitive Interaktion in Virtuellen Umgebungen, etwa durch Körperbewegungen, Gesten oder Spracheingabe. Ihr praktischer Betrieb erweist sich aber als extrem störanfällig. Nicht selten muß bisher ein VR-System neu gestartet werden, weil eines der verwendeten Geräte falsch angeschlossen, nicht richtig konfiguriert oder einfach defekt ist. Die schwierige Handhabbarkeit dieser Geräte macht in vielen Fällen deren potentielle Vorteile zunichte. Im Rahmen dieser Arbeit wird daher ein System zur komfortablen und robusten Anbindung von Interaktionsgeräten vorgestellt.

## 1.2 Aufgabenstellung und Zielsetzung

Im Rahmen dieser Arbeit wird ein Simulationssystem entwickelt, welches besser als die existierenden Ansätze die Realisierung von Virtuellen Umgebungen ermöglicht, die eine große Zahl von bewegten Objekten mit komplexem Verhalten aufweisen welches sowohl statisch als auch dynamisch kann.

Die semantischen Beziehungen zwischen diesen Objekten sollen sowohl statisch definierbar, als auch - durch die Simulation festgelegt - zur Laufzeit veränderbar sein.

Das System muß im Hinblick auf die gestellte Aufgabe effizient, daß heißt für hoch komplexe dynamische Virtuelle Umgebungen mit Reaktionszeiten im Millisekunden-Bereich arbeiten, sodaß Interaktion durch den Benutzer in Echtzeit möglich ist.

Weiter soll das System flexibel erweiterbar und, wo immer möglich, zur Laufzeit konfigurierbar sein.

Benutzerinteraktion erfolgt über multidimensionale Geräte. Dabei soll die Anbindung dieser Geräte hardwareunabhängig und fehlertolerant erfolgen.

Schließlich wird die Anwendbarkeit des Systems anhand von Anwendungen gezeigt werden.

### 1.3 Zusammenfassung der wichtigsten Ergebnisse

Heutige VR-Systeme unterstützen die Entwicklung von sehr komplexen dynamischen Virtuellen Umgebungen nur ungenügend. Die Ursache liegt in der Verwendung des Szenengraphen als Datenbasis für Objektverhalten und in der rein statischen Definition der semantischen Beziehungen zwischen Objekten begründet.

Das im Rahmen dieser Arbeit vorgestellte System zur Simulation von Autonomen Objekten weist diese Nachteile nicht auf und erlaubt die Konstruktion von virtuellen Welten mit mehreren hundert beweglichen Objekten, die dynamisches Verhalten aufweisen und miteinander interagieren. Der Benutzer ist in dieses System integriert und wird in die Lage versetzt, mit den Objekten in seiner Umgebung zu interagieren.

Zentral für die Simulation von Verhalten ist ein innovatives Konzept von Autonomen Objekten und sogenannten Features, die den Autonomen Objekten zur Laufzeit zugeordnet werden können und deren Funktionen erweitern. Das entstandene Simulationssystem ist flexibel einsetzbar, leicht zu erweitern sehr effizient.

Zur Beschreibung der Simulation wird eine neuartige grafische Notation eingeführt, die die Dokumentation von Objektverhalten erlaubt und sich für die visuelle Programmierung von Verhalten in einer grafischen Benutzungsoberfläche eignet.

Im Zusammenhang mit der Entwicklung des Simulationssystems entstanden neue Algorithmen zur Identifizierung von Hindernissen in virtuellen Umgebungen, sowie zur Suche nach kollisionsfreien Pfaden.

Die Erweiterung des Simulationsmodells von der individuellen auf die quantitative Simulation und das Aura-Konzept ermöglicht die Darstellung von Virtuellen Umgebungen mit nahezu unbegrenzter Zahl von autonomen Charakteren.

Die Handhabung von multidimensionalen Interaktionsgeräten wurde im Vergleich zum Stand der Technik wesentlich verbessert.

Nicht funktionierende Geräte können rekonfiguriert oder ausgetauscht werden *während* das VR-System läuft. Ein Neustart ist nicht mehr nötig.

Neuartige Kommunikationsprotokolle reduzieren die Latenz bei der Interaktion wesentlich. Es konnte gezeigt werden, daß die Reduktion maximal

ist.

Durch eine grafische Benutzungsoberfläche, die die Konfiguration, Kalibrierung und den Test von Geräten integriert, wird die Handhabung dieser Geräte wesentlich erleichtert.

Eine erweiterbare und objektorientierte Softwareschnittstelle erleichtert die hardwareunabhängige Integration von multidimensionalen Interaktionsgeräten und schließt auch haptische Geräte mit ein.

## 1.4 Gliederung der Arbeit

Zunächst werden im folgenden Kapitel eine Reihe von existierenden Systemen im Hinblick auf ihre Eignung zur Entwicklung von komplexen dynamischen virtuellen Welten und der Anbindung von multidimensionalen Interaktionsgeräten untersucht. Es wird herausgearbeitet, wo die Probleme im Hinblick auf die Darstellung von Verhalten und die Handhabung von Interaktionsgeräten liegen.

In Kapitel 3 wird ein System zur Simulation von dynamischem Objektverhalten durch Autonome Objekte präsentiert. Die Konzepte und Softwarearchitektur des Simulationssystems werden diskutiert. Auch wird beschrieben, wie das System parallelisiert werden kann.

Kapitel 4 führt zunächst eine grafische Notation zur Beschreibung von Simulationen mit Autonomen Objekten ein. Darauf aufbauend, werden grundlegende Elemente der Simulation beschrieben, anschließend Techniken der Basisinteraktion definiert und schließlich ein allgemeines Benutzermodell eingeführt.

Kapitel 5 befasst sich mit der Integration von multidimensionalen Interaktionsgeräten. Diese werden zunächst klassifiziert. Anschließend werden, aufbauend auf den Erkenntnissen aus Kapitel 2, Forderungen aufgestellt, die ein solches System erfüllen muß. Es folgt eine Beschreibung der Systemarchitektur und der logischen Geräteklassen. Neuartige Protokolle zur Minimierung der Latenz werden vorgestellt. Schließlich wird die grafische Benutzungsoberfläche präsentiert.

In Kapitel 6 wird anhand von drei Anwendungen die Eignung des Systemes zur Lösung der gestellten Aufgabe aufgezeigt.

Die Arbeit schließt mit einer Zusammenfassung.