

1 Einleitung

„Mein Traum ist, Design und Ästhetik mit wirklich praktischem Nutzen zu verschmelzen. Wenn ein Interface nicht schön oder angenehm ist, wollen die Menschen es auch nicht benutzen. Ich möchte die Schnittstelle zwischen Mensch und Computer angenehmer gestalten und dadurch unser Leben bereichern.“

Hiroshi Ishii, MIT Media Lab [TangibleMedia@]

1.1 Hintergrund und Motivation

Seit Jahrtausenden arbeiten Menschen sehr effektiv mit unterschiedlich spezialisierten, mit der Hand (be)greifbaren Werkzeugen und Produkten. Durch die Verwendung von Computern mit ihren an der Desktop-Metapher orientierten *WIMP-Interfaces* (*Windows, Icons, Menus, Pointing Device*) gehen jedoch Anschaulichkeit und Handhabbarkeit im Vergleich zu traditionellen Werkzeugen verloren. Es verwundert also nicht, daß trotz der Erfolgsgeschichte der Desktop-Metapher die Nutzung fensterbasierter Betriebssysteme nach wie vor von vielen – insbesondere unerfahrenen – Nutzern als nur wenig natürlich empfunden wird. Das liegt unter anderem an den zahlreichen stereotypen Mauspositionierungs- und Klick-Handlungen bei der Computerbedienung oder aber an der häufig beschwerlichen Suche von Funktionen innerhalb uniform aussehender, nur durch Texte differenzierbarer Menüeinträge. So wird bei gegenwärtigen 2D-Benutzungsschnittstellen menschliche Handlungseffizienz verschenkt, da Menschen in der Realität an viel komplexere Suchräume gewöhnt sind, eine sehr gute räumliche Orientierung besitzen und äußerst erfolgreich mit gestaltorientierten Werkzeugen arbeiten können.

Interaktive dreidimensionale Computergrafik als Bestandteil von *Post-WIMP-Interfaces* besitzt jedoch ein inhärentes Potential, diese menschlichen Fähigkeiten effektiver zu nutzen. Erkenntnisse aus der Gestaltung und dem Gebrauch realer Werkzeuge können für *dreidimensionale Benutzungsschnittstellen* (kurz: *3D-GUIs*) in den virtuellen 3D-Raum übertragen werden. Dabei entstanden 3D-GUIs bisher vor allem bei Forschungsarbeiten auf dem Gebiet der Virtuellen Realität (VR), die jedoch lange Zeit sehr auf Hardware-Technologieentwicklung fokussiert waren. Obwohl in den 90er Jahren bereits mehrere Arbeiten zu 3D-Interaktionstechniken und *3D-Widgets* – also Interaktionselementen eines 3D-GUIs – innerhalb von VR-Projekten durchgeführt wurden, gibt es zur 3D-Anwendungsentwicklung nach wie vor kaum Richtlinien und keine Standards. Diese sind jedoch laut Shneiderman eine wichtige Säule erfolgreichen User-Interface-Designs [Shnei98]. Im Gegensatz zu klassischer WIMP-Softwareentwicklung mangelt es im 3D-Bereich zudem an Designempfehlungen, wiederverwendbaren Bausteinen und geeigneten Toolkits.

Eine Ursache dafür ist der geringe Verbreitungsgrad von VR-Anwendungen, die häufig auf teure Hardwarelösungen in bestimmten Branchen ausgerichtet und damit zumeist Experten vorbehalten sind. Eine lohnenswerte Alternative, die mit der dynamischen Leistungsentwicklung von Prozessoren und Grafiksubsystemen im Massenmarkt der PCs und Spielekonsolen zunehmend in den Mittelpunkt rückt, stellt die sogenannte *Desktop-VR* dar. Das ist interaktive 3D-Grafik ohne komplizierte und teure Spezialhardware unter Nutzung von Standard-PCs. Gegenüber aufwendigen VR-Lösungen bieten Desktop-VR-Anwendungen einige Vorteile und eröffnen neue Anwendungsdomänen. Dabei erleichtern 3D-Widgets als Mittler zwischen niedrigdimensionalen Eingabegeräten (wie Tastatur und Maus) und höherdimensionalen Auf-

gaben im 3D-Raum (z.B. Positionierung) die Arbeit mit Desktop-VR-Anwendungen. Anders als z.B. textuelle Menüeinträge besitzen sie zudem in virtueller Form erneut Werkzeugcharakter und versprechen damit mehr Anschaulichkeit. Im Vergleich zum klassischen VR-Sektor gibt es im Bereich Desktop-VR bisher jedoch nur wenige Forschungsarbeiten zum Aufbau und zur Erstellung von 3D-Anwendungen und 3D-GUIs.

Erfolgreich werden 3D-Anwendungen vielleicht nicht – wie ursprünglich angenommen – in Form von 3D-Betriebssystemoberflächen als Substitution der Desktop-Metapher sein. Für zahlreiche andere Bereiche stellen sie jedoch lohnenswerte Alternativen dar, z.B. bei 3D-Modellierungs- und Animationsprogrammen oder kollaborativen Arbeitsumgebungen. Besonders deutlich wird der Nutzen aber für die sehr große Gruppe der Alltagsnutzer, wenn es sich um interaktive 3D-Grafikanwendungen im World Wide Web (kurz: *Web3D-Grafik*) handelt. Der PC wird zunehmend als VR-Plattform der Zukunft betrachtet, wobei das Internet hierbei eine entscheidende Rolle spielen wird [Bulli98]. Anwendungsdomänen mit Erfolg sind u.a. Produktpräsentationen, E-Commerce-Anwendungen, verteilte Mehrbenutzerumgebungen oder Architekturvisualisierungen.

Auch große Firmen, wie Adobe und Macromedia, leisten inzwischen mit proprietären Web3D-Formaten einen Beitrag dazu, das Spektrum interaktiver multimedialer Web-Anwendungen zu bereichern. Parallel dazu betreiben Institutionen – wie das *Web3D-Consortium* und die *Moving Picture Experts Group* – die Entwicklung von internationalen Standards, welche Sicherheit, Austauschbarkeit, Unabhängigkeit und Interoperabilität zu anderen Standards (SVG, SMIL etc.) garantieren sollen. Trotz zahlreicher Web3D-Formate gibt es jedoch keinen etablierten Autorenprozeß, häufig unzureichende Autorenwerkzeuge und nur wenige Konzepte zur einfachen Wiederverwendung existierender 3D-Bestandteile. Die meisten Anwendungen werden *from scratch* mit viel Expertenwissen und Programmiererfahrung erstellt, was lange Entwicklungszeiten und Handarbeit einschließt. Während für „klassische“ Medien bereits ausgereifte Werkzeuge und langjährige Erfahrungen auch in multidisziplinären Teams existieren, besteht hier für Web3D-Grafik noch deutlicher Nachholbedarf.

Vision

Vor dem Hintergrund der aktuellen Performance-Verbesserungen im Grafikhardwarebereich, der Verbesserung der Netzzugänge und der künftigen Entwicklung skalierbarer, adaptiver, multimedialer und interaktiver Web-Anwendungen ist die Herausforderung darin zu sehen, Web3D-Grafik als wichtigen und gleichberechtigten Teil dieser Entwicklungen zu etablieren. Dazu soll mit dieser Arbeit ein Beitrag geleistet werden. Die Vision besteht darin, in einfacher Weise 3D-Anwendungen aus wiederverwendbaren, standardisierten Bausteinen zusammenzusetzen, wobei visuelle High-Level-Werkzeuge und ein intuitiver Autorenprozeß zum Einsatz kommen. Daran sollten Experten mit verschiedenen fachlichen Hintergründen beteiligt sein, also neben Programmierern z.B. auch Designer und Soundexperten. Weniger imperative und primär visuelle Programmierung würde auch einem breiten Personenkreis die Erstellung derartiger Anwendungen ermöglichen. Schließlich sollen neben der technischen Basis, die sich auf offene Standards gründen muß, auch der metaphorische Teil und die Benutzungsschnittstellen von 3D-Anwendungen weiter entwickelt werden. Da der technische Rahmen allein den Erfolg von 3D-Anwendungen nicht garantiert, müssen Richtlinien zur Erstellung interaktiver 3D-Applikationen, Designregeln und ein Repertoire von 3D-Interaktionstechniken und 3D-Widgets künftigen Entwicklern zur Verfügung stehen. Mit einem Zitat aus den Webseiten von Microsoft Research [MicroResearch@] soll die einleitende Motivation abgeschlossen werden.

„We believe that interactive 3D graphics and animation will be an important component of future user interfaces directed towards business, consumer and entertainment applications.“

1.2 Problemdefinition und Zielsetzung

Im folgenden soll dargestellt werden, welche konkreten Probleme im skizzierten Themenspektrum existieren und was für Arbeitsthesen und Forschungsziele sich daraus ergeben haben. Folgende Defizite und Beobachtungen lassen sich stichpunktartig aufführen:

- 3D-Widgets sind bisher unzureichend erforscht worden, nur ein relativ kleines Repertoire an 3D-Widgets existiert [Dam92], [Geige98], [Pierc01]. Besonders bei Desktop-VR gibt es kaum Systemisierungsansätze und Standardisierungsbemühungen [Leine97].
- Der Entwicklungsaufwand für 3D-Anwendungen ist zu hoch, es gibt kaum Wiederverwendungskonzepte und wenig Flexibilität. Zu viele Programmierkenntnisse sind nötig.
- 2D-Schnittstellen sind für den 3D-Raum größtenteils ungeeignet, oft erfolgte achtlose und wenig effektive Übertragung von 2D-Metaphern (Designproblematik der 3. Dimension).
- Nur selten erfolgt eine interdisziplinäre Entwicklung von 3D-Anwendungen, die aufgrund der Komplexität des Designs in der dritten Dimension jedoch dringend erforderlich wäre.
- 3D-Formate bieten selten High-Level-Ansätze oberhalb des Szenengraphniveaus.
- Entwicklungsunsicherheit und Formatabhängigkeit existieren durch die Vielfalt proprietärer und standardisierter Web3D-Formate. Die Entwicklung von X3D und MPEG-4 XMT sind noch nicht abgeschlossen; 3D-Grafik ist bisher im Web noch nicht richtig etabliert.
- Die Definition von 3D-Sound und die Integration weiterer Medien wurde bisher unzureichend adressiert. Zudem wurde die Realisierung von Interaktivität und Objektverhalten in Web3D-Formaten bisher ungenügend unterstützt oder ist nur eingeschränkt und mit externer Programmierung möglich.
- Als Trend im Web- und Multimediabereich läßt sich die Entwicklung von programmzentrierten hin zu dokumentenzentrierten Anwendungen beobachten. Multimediale Formate werden zunehmend auf XML-Basis entwickelt.

Thesen

Aus der Einordnung der Forschungsthematik und der Betrachtung der Defizite lassen sich folgende Arbeitsthesen ableiten, die auch für die gesamte Arbeit als Motivation zu sehen sind.

- Desktop-VR-Schnittstellen können effektiver und benutzungsfreundlicher als aufwendige VR-Lösungen sein, wenn die Komplexität des 3D-Raumes bezüglich Erscheinung und Interaktion im technischen und Gestaltungssinne beherrscht wird.
- Der bisher geringe Erfolg von interaktiven 3D-Grafikanwendungen ist nicht auf ungenügende Hardwarefähigkeiten, sondern primär auf kaum vorhandene Designrichtlinien, fehlende Standardisierungsbemühungen, ungenügende Wiederverwendungsmöglichkeiten und mangelnde Werkzeugunterstützung zurückzuführen.
- Erst wenn 3D-Anwendungen für weit verbreitete Zielplattformen schnell und komfortabel zu entwickeln sind, lassen sich Benutzbarkeitsstudien für 3D-Grafik durchführen und ihre Eignung für konkrete Anwendungsdomänen, Zielgruppen und Endgeräte sicher beurteilen. (Bei [Myers00] wird zudem vorausgesagt, daß erst eine Reihe etablierter 3D-Interaktionstechniken und 3D-Widgets existieren muß, bevor High-Level-Werkzeuge entworfen und sinnvolle 3D-Applikationen auf breiterer Basis entwickelt werden können.)
- Flexible deklarative Dokumentenformate sind im Zusammenhang mit visuellen 3D-Autorenwerkzeugen besser für die Entwicklung multimedialer dreidimensionaler Anwendungen geeignet als APIs. Nur so können erfolgreiche Anwendungen kostengünstig für breite Nutzerkreise entwickelt werden.

Zielsetzung

Vor diesem Hintergrund ist die Zielstellung dieser Arbeit zu sehen, d.h. die einfache, komponentenbasierte Entwicklung intuitiver und attraktiver dreidimensionaler Anwendungen mit Hilfe eines 3D-Autorenwerkzeuges. Dabei liegt die Vision zugrunde, ein standardisiertes Repertoire von wiederverwendbaren, anpaßbaren, frei verfügbaren oder kommerziell vertriebenen 3D-Schnittstellenelementen (3D-Widgets) zur interdisziplinären Konstruktion von 3D-Anwendungen zu nutzen. Hieraus ergaben sich im einzelnen die folgenden Forschungsziele.

- Leistung eines Beitrages zur grundlegenden Klassifizierung und künftigen Standardisierung von 3D-Schnittstellenelementen und Vergrößerung des verfügbaren Repertoires.
- Entwicklung von Metaphern und Gestaltungsrichtlinien für 3D-Benutzungsschnittstellen.
- Entwicklung eines komponentenorientierten, von Toolkits unabhängigen 3D-Dokumentenmodells auf XML-Basis sowie der Rahmenarchitektur zur Verarbeitung der deklarativen Beschreibungsdokumente.
- Deklarative Spezifikation von einzelnen 3D-Widgets inklusive ihres Verhaltens und multimedialer Bestandteile sowie Beschreibung der Konfiguration, Assemblierung und des Zusammenwirkens von Komponenten in 3D-Anwendungen mit Hilfe des von Szenengraphen abstrahierenden 3D-Dokumentenmodells.
- Konzeption und prototypische Entwicklung eines visuellen 3D-Autorenwerkzeuges mit Unterstützung verschiedener Autorenrollen und Überprüfung des Autorenprozesses an 3D-Beispielanwendungen.
- Nutzung von Standards, Internetfähigkeit, Flexibilität des Dokumentenmodells und des Autorenwerkzeuges für künftige Erweiterungen und neue Systemplattformen, Übersetzung der erstellten Anwendungen in 3D-Zielformate.

Abgrenzung

Die vorliegende Arbeit steht im Schnittpunkt verschiedener Fachgebiete der Informatik. Die *Computergrafik* spielt dabei mit Szenengraph-basierten Formaten für interaktive Web3D-Anwendungen eine wichtige Rolle, wobei jedoch weder ein neues Format noch ein Player entwickelt werden sollten, sondern eine Abstraktionsebene oberhalb von Szenengraphen.

Aus den Bereichen *Human Computer Interaction*, *Usability* und *Design* lassen sich zahlreiche Erkenntnisse auch für die Gestaltung von dreidimensionalen Benutzungsoberflächen ableiten. Auch wenn die Gestaltung ein wichtiges Element für den Erfolg von 3D-Anwendungen darstellt, nimmt sie jedoch nur einen Teil der Arbeit ein.

Die entwickelte Komponentenarchitektur und prototypische Realisierung legen Bezüge zur *Softwaretechnologie* nahe. Da aber deklarative Dokumentbeschreibungssprachen auf XML-Basis und ein werkzeuggestützter Autorenprozeß konzipiert wurden, ist der primäre Bezug zur *Multimediatechnik* mit ihrem Teilgebiet der multimedialen Dokumentformate für das World Wide Web zu sehen. In dieser Arbeit werden jedoch weder Server-Architekturen für Web3D-Grafik, noch Streaming-Möglichkeiten für die Übertragung zum Client-Rechner betrachtet, sondern High-Level-Beschreibungsformate und der damit verbundene Autorenprozeß sowie grundlegende Metaphern und Bestandteile dreidimensionaler Desktop-VR-Applikationen.

1.3 Aufbau der Arbeit

Anhand der Zielsetzung wird deutlich, daß die Arbeit sich in zwei wesentliche Teile bzw. Hauptbeiträge gliedert. Das sind

- die Klassifikation von 3D-Metaphern und 3D-Widgets für dreidimensionale Benutzungsschnittstellen und
- die Entwicklung eines deklarativen Komponentenansatzes und intuitiven Autorenprozesses zur Erstellung von interaktiven 3D-Anwendungen.

Das Kapitel 3 ist dem ersten Teil gewidmet. Mit der systematischen Analyse von Widgets und der Bereitstellung eines Repertoires an einheitlich spezifizierten und realisierten 3D-Widgets wird ein Beitrag zu künftigen Standardisierungen geleistet. Nach diesem theoretisch-konzeptionellen Teil wird in den folgenden drei Kapiteln dafür die komponentenbasierte Architektur vorgestellt, die jenseits der Spezifikation von Widgets ihre Wiederverwendbarkeit in verschiedenen 3D-Applikationen erst möglich macht. Dabei erfolgt die Vorstellung des Lösungsansatzes in Form der 3D-Komponentenarchitektur (Kapitel 4), des deklarativen Dokumentenmodells (Kapitel 5) und des Autorenprozesses mit seinen Werkzeugen (Kapitel 6).

Aufteilung der Kapitel

Kapitel 2 führt in das Gebiet der 3D-Echtzeitgrafik im World Wide Web ein, umreißt das Forschungsfeld und die damit verbundenen Probleme. Anhand der Vorstellung von Anwendungsgebieten und Beispielen wird deutlich, welches Potential Web3D-Grafik besitzt. Die sich anschließende Betrachtung von Web-basierten 3D-Formaten und Technologien macht die Trennung in die beiden Lager der proprietären und Standardformate deutlich. Schließlich werden aktuelle Trends und Erfolgsfaktoren in diesem Sektor besprochen. So steht die technische Basis zwar in hinreichendem Maße zur Verfügung, es existieren jedoch Defizite bei der einfachen Erstellung von Web3D-Grafik, dem Aufbau von 3D-Anwendungen und ihrer intuitiven Bedienung.

Kapitel 3 beschäftigt sich daher mit Metaphern und Widgets für interaktive 3D-Anwendungen und stellt das theoretische Fundament dieser Arbeit dar. Hierbei werden vor allem Ergebnisse aus dem Bereich der VR-Forschung einbezogen, die sich jedoch teilweise auf den jungen Bereich Desktop-VR (und damit auch Web3D-Grafik) übertragen lassen. 3D-Benutzungsoberflächen als Bestandteil von 3D-Anwendungen werden charakterisiert, mit Beispielen vorgestellt und mit 2D-GUIs verglichen. Gestaltungsrichtlinien für 3D-Anwendungen werden entwickelt. Zur räumlichen Gliederung von 3D-Anwendungen wird das Konzept der *Action Spaces* eingeführt, innerhalb derer sich 3D-Objekte mit 3D-Widgets bearbeiten lassen. Schwerpunkt des Kapitels ist jedoch die Systematisierung und Beschreibung von 3D-Widgets als Interaktionselemente von 3D-GUIs.

Kapitel 4 stellt die Konzeption der 3D-Komponentenarchitektur CONTIGRA (*Component-Oriented Three-dimensional Interactive GRaphical Applications*) als möglichen Lösungsansatz für die im 2. und 3. Kapitel genannten Probleme vor. Die Notwendigkeit der exakten Spezifikation, flexiblen Anpaßbarkeit und einfachen Kombinierbarkeit von 3D-Widgets kann als Motivation für eine Architektur gesehen werden, für die zunächst eine Anforderungsanalyse vorgenommen wird. Als Realisierungsbasis kommen Komponentenmodelle aus der Softwaretechnik und spezialisierte 3D-Komponentenmodelle in Frage. Diese verwandten Arbeiten werden detailliert betrachtet, bevor der eigene Komponentenbegriff und dokumentenzentrierte Lösungsansatz im Gesamtüberblick vorgestellt und dessen Notwendigkeit begründet wird. Dazu zählen die einzelnen Entwicklungsebenen für CONTIGRA-Komponenten und die entwickelten deklarativen Beschreibungssprachen auf XML-Basis.

Kapitel 5 stellt das deklarative Dokumentenmodell des CONTIGRA-Ansatzes im Detail vor. Dafür kommt die *Extensible Markup Language (XML)* zum Einsatz. In drei Unterkapiteln folgt die Beschreibung der entwickelten XML-Schemata *CoApplication*, *CoComponent* und *CoComponentImplementation*. Auf der obersten Ebene ist *CoApplication* als Beschreibungssprache für interaktive 3D-Applikationen angesiedelt, mit der sich typische Szenenparameter festlegen lassen. Jede CONTIGRA-Komponente besteht aus einem Schnittstellendokument mit Parametern und Metainformationen (*CoComponent*) und einem Implementierungsdokument (*CoComponentImplementation*). Innerhalb der Implementierung gibt es die drei Szenengraphen *Geometry*, *Audio* und *Behavior*. Während für den Geometrieteil X3D zum Einsatz kommt, werden für die anderen Teile die eigens entwickelten Sprachen *Audio3D* und *Behavior3D* verwendet und im Überblick vorgestellt.

Kapitel 6 behandelt schließlich den Autorenprozeß und seine Werkzeuge für die CONTIGRA-Komponentenarchitektur. Dazu wird zunächst eine Analyse existierender 3D-Autorenwerkzeuge aus dem Web3D-Bereich durchgeführt. Der CONTIGRA-spezifische Autorenprozeß wird anschließend mit seinen Phasen und beteiligten Personen erläutert. Daraus leiten sich Anforderungen an ein unterstützendes 3D-Autorenwerkzeug ab. Der CONTIGRABUILDER wird als prototypische Realisierung im Überblick und aus Nutzersicht dargestellt. Es schließen sich Aspekte der Realisierung des Prototypen an. Dazu zählt auch die Transformation der CONTIGRA-Komponenten und -Applikationen in konkrete Zielformate, womit die praktische Umsetzbarkeit des vorgestellten Ansatzes demonstriert wird. Abschließend rundet die Vorstellung von Beispielanwendungen das Kapitel ab.

Kapitel 7 ist schließlich eine Zusammenfassung der gesamten Arbeit, wobei ihr Beitrag zum Forschungsgebiet und ihre Limitationen betrachtet werden. Ein Ausblick beschreibt am Ende, wie der vorgestellte Ansatz fortgeführt werden kann und welche interessanten Forschungsarbeiten sich künftig daraus ergeben.