



Eine XML-basierte Komponentenarchitektur für interaktive, multimediale 3D-Anwendungen

An XML-based Component Architecture for Interactive Three-dimensional Multimedia Applications

Raimund Dachzelt, Technische Universität Dresden, Nominierung GI-Dissertationspreis 2004¹

Zusammenfassung Für die Entwicklung interaktiver 3D-Grafikanwendungen existieren bisher kaum Standards. Mit den in der Dissertation vorgestellten Gestaltungsrichtlinien und Metaphern für DesktopVR-Benutzungsschnittstellen sowie einer Systematisierung existierender 3D-Interaktionselemente soll dem Entwickler die Konzeption von 3D-Anwendungen erleichtert werden. Für deren Erstellung wird eine dokumentenzentrierte Komponentenarchitektur auf XML-Basis vorgestellt. Mit Hilfe eines visuellen Autorenwerkzeuges ist es nun möglich, interaktive 3D-Anwendungen aus wiederverwendbaren, standardisierten Bausteinen zusammenzusetzen und in verschiedene

3D-Zielformate, vor allem für das World Wide Web, zu transformieren. ▶▶▶ **Summary** Despite research on Virtual Reality user interfaces there exists no standard for developing 3D applications. The PhD thesis proposes design guidelines and metaphors especially for DesktopVR UIs and provides a classification and specification approach for 3D widgets. A document-centered component architecture on the basis of XML along with a visual authoring tool allows the implementation of interactive 3D applications consisting of reusable, standardized components. The developed applications can be transformed into various 3D target formats, mainly for the World Wide Web.

KEYWORDS H.5.2 [Information Interfaces], I.3 [Computer Graphics], I.3.6 [Methodology and Techniques], I.3.7 [Three-Dimensional Graphics and Realism] User Interfaces, Graphical User Interfaces (GUI), Prototyping, Standardization; Virtual Reality

1 Einleitung und Motivation

Obwohl gegenwärtige, an der Desktop-Metapher orientierte Benutzungsschnittstellen omnipräsent

¹ Die Dissertation von Herrn Dr. Dachzelt wurde für den GI-Dissertationspreis 2004 vorgeschlagen und erhielt einen Dissertationspreis der Commerzbank-Stiftung 2005. Die Gutachter der an der Technischen Universität Dresden durchgeführten Promotion waren Prof. Dr.-Ing. Klaus Meißner (Lehrstuhl für Multimedialechnik, TU Dresden), Prof. Dr.-Ing. Dr.-Ing. E.h. Wolfgang Strasser (Wilhelm-Schickard-Institut Graphisch Interaktive Systeme, Universität Tübingen) und Prof. Dr. Oliver Deussen (Lehrstuhl Computergrafik & Medieninformatik, Universität Konstanz).

und erfolgreich sind, versprechen 3D-Benutzungsoberflächen (3D-GUIs) in bestimmten Bereichen ein intuitiveres Arbeiten, indem sie unter anderem die menschliche Fähigkeit zur Raumorientierung besser nutzen. 3D-GUIs wurden bisher jedoch vor allem im Bereich Virtuelle Realität (VR) entwickelt, wo trotz langjähriger Forschungsarbeiten noch keine Entwicklungsstandards existieren. Das trifft erst recht auf den Bereich der Desktop-VR zu, also interaktiver 3D-Grafik unter Nutzung von Standard-PCs.

Dabei erhält durch die Leistungsentwicklung von Prozessoren und Grafiksensoren sowie breitbandige Internet-Anbindungen erstmals ein großer Nutzerkreis Zugang zu Web-basierten 3D-Anwendungen. Hier sollen als Beispiele nur interaktive Produktkonfiguratoren in E-Commerce-Anwendungen oder interaktive Geovisualisierungen genannt werden. 3D-Echtzeitgrafik im World Wide Web (kurz Web3D) steht als wichtiger Teilbereich der Desktop-VR im Fokus der Dissertation [1].

Defizite existieren nach wie vor bei einem intuitiven, werkzeuggestützten Autorenprozess solcher Anwendungen sowie in Form hoher Erstellungskosten aufgrund mangelnder High-Level-Ansätze zur Wiederverwendbarkeit. Die Formatvielfalt im Web3D-Bereich erzeugt außerdem Entwicklungsunsicherheit und Abhängigkeiten, zudem ist die Definition von 3D-Sound und die Integration weiterer Medien bisher kaum adressiert worden. Auch die Realisierung von Interaktivität und Objektverhalten in Web3D-Formaten ist bisher nur eingeschränkt möglich. Vor diesem Hintergrund war es das Ziel dieser Arbeit, interaktive und multimediale 3D-Anwendungen in einfacher Weise aus einem Repertoire wiederverwendbarer, standardisierter Bausteine zusammensetzen zu können. Die Dissertation gliedert sich in zwei wesentliche Hauptbeiträge, die in den folgenden Kapiteln näher beschrieben werden.

2 Metaphern und Widgets für interaktive 3D-Anwendungen

Der erste Teil der Dissertation widmet sich Metaphern und Interaktionselementen für interaktive 3D-Anwendungen im Bereich Desktop-VR und stellt damit das theoretische Fundament der Arbeit dar. Wesentliche Hauptbestandteile einer 3D-GUI sind die zu bearbeitenden 3D-Dokumente sowie die integrierten Interaktionselemente zu ihrer Manipulation. Diese so genannten *3D-Widgets* erleichtern als Mittler zwischen niedrigdimensionalen Eingabegeräten und höherdimensionalen Aufgaben die Interaktion im 3D-Raum. Der Nutzen von 3D-GUIs im Vergleich zu traditionellen Interfaces, z. B. für Betriebssystemoberflächen oder kollaborative Arbeitsumgebungen, wurde bisher kontrovers diskutiert. In der Dissertation konnte gezeigt werden, dass für 3D-Anwendungen zunächst Fortschritte auf konzeptioneller und Implementierungsseite nötig sind, bevor sich

die Effektivität von 3D-GUIs für bestimmte Anwendungsdomänen zuverlässig überprüfen lässt. Den Vorteilen von 3D-GUIs, wie räumlichem Orientierungs- und Handlungswissen, verbesserter Aufmerksamkeitssteuerung und höherer Attraktivität stehen z. B. Navigations- und Interaktionsprobleme gegenüber. Einige der dafür in der Arbeit entwickelten Lösungsansätze werden nachfolgend zusammengefasst.

2.1 Metaphern zur Raumgliederung

Das entwickelte Konzept der *Action Spaces* bettet 3D-Dokumente und -Widgets als Hauptbestandteile von 3D-Anwendungen in räumliche Strukturen ein. *Action Spaces* sind virtuelle dreidimensionale Räume bzw. aufgabenzentrierte Zonen, zwischen denen Nutzer einfach navigieren können. 3D-Widgets werden darin zusammen mit ihren Containern um einen vordefinierten Blickpunkt gruppiert, womit sie Arbeitsplätzen aus der realen Welt ähneln. Mit der effektiven Aufteilung des Bildschirms durch vordefinierte Anordnungen und Blickpunkte, der Integration von Widgets in Raumstrukturen und der vereinfachten Navigation zwischen einzelnen Bereichen der 3D-Anwendung werden wichtige Probleme gerade für Desktop-VR-Anwendungen gelöst.

2.2 Klassifikation und Spezifikation von 3D-Widgets

Ein Hauptfokus lag auf der systematischen Klassifikation und Spezifikation von 3D-Widgets als zentralem Bestandteil von Desktop-VR-Anwendungen. Der Mangel an frei verfügbaren Widgets als Grundlage möglicher Standards sowie ungenügende Ansätze zu ihrer Systematisierung und Klassifikation motivierten diesen Teil der Arbeit. Im Gegensatz zu bereits existierenden Taxonomien für reine VR-Interaktionstechniken lag der Schwerpunkt der eigenen Klassifikation auf interaktiven 3D-Widgets mit geometrischer Repräsentation. Die Klassifikation

der Widgets erfolgte nach dem Kriterium des Einsatzbereiches bzw. Interaktionszieles und orientiert sich somit an praktischen Bedürfnissen von 3D-Applikationsentwicklern. Es wurden sowohl aus 2D-GUIs übernommene als auch speziell für den 3D-Raum entwickelte Interaktionselemente einbezogen und in eine umfassende Taxonomie eingeordnet. Eine ausführliche Darstellung der Kategorien und der jeweils zugehörigen Widgets findet sich in der Dissertation [1] und in interaktiver Form online [2]. Ein besonderer Fokus lag auf Techniken zur Anwendungskontrolle und dabei speziell auf dreidimensionalen Menülösungen, für die eine umfassende Übersichtsdarstellung und eine eigene Taxonomie entwickelt wurden [3].

Die zahlreichen betrachteten Widgets und Interaktionstechniken wurden nicht nur klassifiziert, sondern auch in einheitlicher Form mit ihren Parametern und ihrer Funktionalität spezifiziert. Erst dadurch wird ihre applikationsübergreifende (Wieder-)Verwendung und flexible Kombination im Sinne der künftigen Standardisierung von 3D-GUIs möglich. Eine Auswahl aus dem verfügbaren Repertoire ist in Bild 1 zu sehen.

3 CONTIGRA – eine deklarative 3D-Komponentenarchitektur

Im zweiten Teil der Dissertation wurde die 3D-Komponentenarchitektur Contigra (*Component-Oriented Three-dimensional Interactive GRaphical Applications*) als praktischer Lösungsansatz zur Spezifikation von 3D-Widgets und zur Erstellung interaktiver, multimedialer 3D-Anwendungen vorgestellt. Die in [1] detailliert betrachteten verwandten Arbeiten weisen Abhängigkeiten von konkreten 3D-Formaten, Technologien, Programmiersprachen oder Realisierungsplattformen auf und bieten keine High-Level-Abstraktion oberhalb des Szenengraphs, einer innerhalb der Computergrafik gebräuchlichen Form der Szenenrepräsentation.

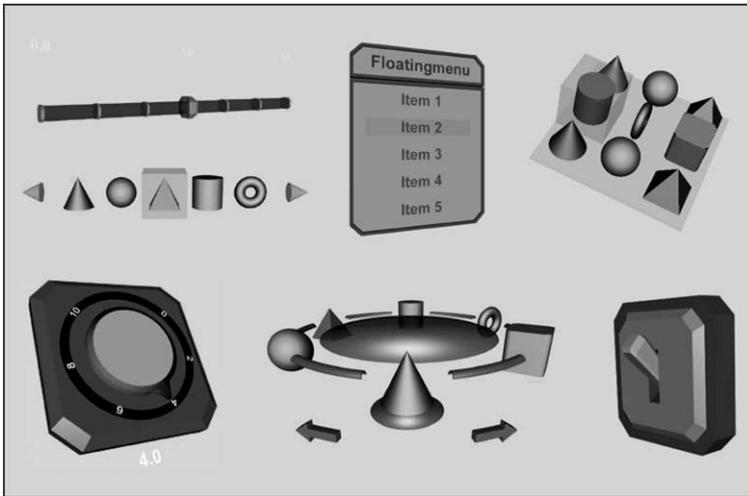


Bild 1 Auswahl spezifizierter und implementierter 3D-Widgets zur Anwendungskontrolle.

3.1 Konzeption der Komponentenarchitektur

Für die Erstellung interaktiver 3D-Grafikapplikationen wurde ein eigener dokumentenzentrierter Komponentenansatz vorgeschlagen. 3D-Anwendungen und Komponenten sind dabei ausschließlich durch strukturierte Dokumente beschrieben, mit denen sich von der Schnittstelle über die Komponentenimplementierung bis hin zur Konfiguration, Komposition und Verknüpfung alle Details konsequent deklarativ beschreiben lassen. Bild 2 zeigt die Entwicklungsebenen für Komponenten der Contigra-Architektur, die damit verbundenen

Aufgaben, zugehörigen Dokumente und verwendeten Werkzeuge.

Die *Komponentenentwicklungsebene* umfasst die Implementierung von 3D-Komponenten und die Spezifikation ihrer Schnittstelle für den Zugriff auf die Parameter der Komponente. Die Implementierung unterscheidet sich dabei insofern von der imperativen Softwareprogrammierung, dass die Beschreibung von Geometrie, Erscheinungsbild, Verhalten und auditiven Eigenschaften deklarativ erfolgt. Die *Distributions-ebene* umfasst die Suche, Auswahl und Beschaffung von 3D-Komponenten über das Web, wofür unter [2] ein prototypisches Portal

existiert. Auf der Ebene der *Anwendungsentwicklung* erfolgt zunächst die Anpassung und Konfiguration ausgewählter Komponenten. Diese werden dann zu formatunabhängigen 3D-Anwendungen oder komplexeren Komponenten zusammengefügt und in Form verbindungsorientierter deklarativer Programmierung miteinander verknüpft. Aus den einzelnen Beschreibungsdokumenten kann im Autorenwerkzeug ContigraBuilder oder über externe Transformationen schließlich eine auf der *Laufzeitebene* ausgeführte Applikation für ein konkretes 3D-Zielformat generiert werden. Durch die Trennung von Komponenten- und Anwendungsentwurf in Form verschiedener Dokumente wird Wiederverwendbarkeit explizit unterstützt.

3.2 Die deklarativen Beschreibungssprachen

Das Herz des Contigra-Ansatzes stellen die auf der Basis der *Extensible Markup Language (XML)* entwickelten Auszeichnungssprachen für die einzelnen Komponentenebenen dar. Auf der obersten Ebene ist eine Beschreibungssprache für die gesamte Anwendung realisiert worden, mit der sich typische Szenenparameter festlegen lassen. Für einzelne 3D-Komponenten existie-

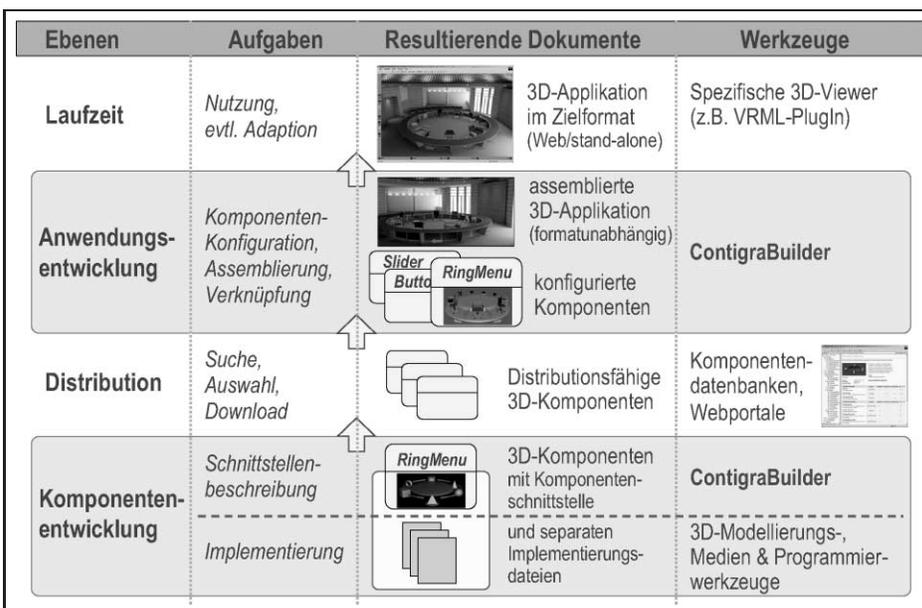


Bild 2 Komponentenebenen der Contigra-Architektur.

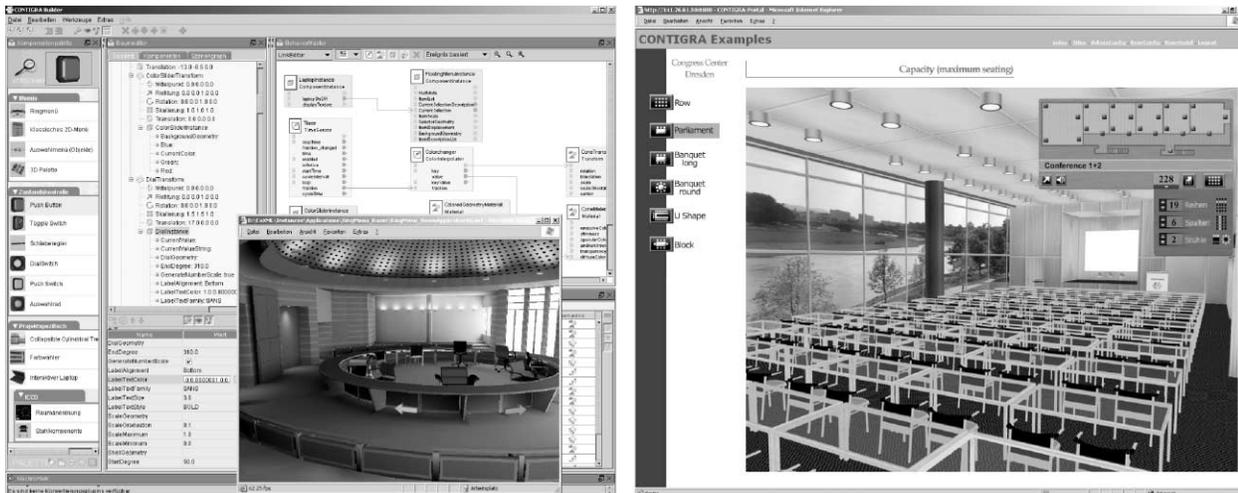


Bild 3 3D-Autorenwerkzeug ContigraBuilder (links) und generierte VRML97-Beispielanwendung eines interaktiven virtuellen Kongresszentrums (rechts).

ren zwei XML-Schemata, da jede Contigra-Komponente aus einem Schnittstellendokument mit konfigurierbaren High-Level-Parametern und Metainformationen sowie einem Dokument zur Beschreibung ihrer Implementierung besteht.

Die Implementierung umfasst eine Transformationshierarchie von Subkomponenten, drei getrennte Hierarchien für *Geometry*, *Audio* und *Behavior* sowie Verknüpfungsinformationen für alle Bestandteile. Während für den Geometrieteil der Web3D-Standard X3D zum Einsatz kommt, werden für die anderen Teile als multimediale Erweiterung die eigens entwickelten Sprachen *Audio3D* [4] und *Behavior3D* [5] verwendet. Mit *Audio3D* können komplexe akustische Szenen (Soundquellen, Hinderisobjekte, Räume, Hallparameter usw.) unabhängig von 3D-Sound-APIs beschrieben werden, wobei die Deklaration spezialisierter Soundquellen und verbundener Räume wichtige Neuerungen darstellen. *Behavior3D* ist ein flexibles, objektorientiertes Konzept zur primär deklarativen Modellierung des Verhaltens von 3D-Objekten und Interaktionen in Anlehnung an SMIL-Konstrukte.

3.3 Das Autorenwerkzeug ContigraBuilder und Beispielanwendungen

Um 3D-Komponenten bzw. 3D-Anwendungen erstellen und damit

Instanzdokumente der Contigra-Schemata visuell bearbeiten zu können, wurde das Autorenwerkzeug *ContigraBuilder* konzipiert und prototypisch in Java realisiert (siehe Bild 3). Die mit dem Werkzeug erstellten XML-Dokumente können durch Objektmodell-gestützte oder auf XSLT-Stylesheets basierende Transformationen flexibel in verschiedene Zielformate überführt werden. Realisiert sind bereits Transformationen nach HTML (für das Komponentenportal), VRML97, X3D und OpenSG sowie teilweise nach Java3D und MPEG-4. Die praktische Umsetzbarkeit des Ansatzes wurde an mehreren Beispielen demonstriert. Dazu zählt eine komplexe Web3D-Anwendung zur Planung von Raumlayouts für ein internationales Kongresszentrum (Beispielansicht im Bild 3 rechts). Deren Realisierung mit mehreren hundert Komponenteninstanzen und zahlreichen Interaktionsmöglichkeiten zeigt die Skalierbarkeit des Contigra-Ansatzes, der vor allem für die Anwendungsklassen Produktpräsentationen/E-Commerce, Lehr-/Lernanwendungen sowie Informationsvisualisierung und Navigation geeignet ist.

4 Zusammenfassung

Die Dissertation leistet mit den für den Bereich Desktop-VR entwickelten Gestaltungsrichtlinien, einem Konzept zur metaphorbasier-

ten Gliederung virtueller Räume sowie der Klassifikation und Spezifikation zahlreicher 3D-Widgets, einen Beitrag zur Weiterentwicklung und Standardisierung von 3D-GUIs. Damit steht ein systematisiertes Repertoire an wiederverwendbaren Interaktionsbausteinen zur Verfügung, die sich anpassen, zusammenfügen und zu 3D-Anwendungen verknüpfen lassen. Dafür wurde ein durchgängig deklaratives Dokumentenmodell auf XML-Basis samt visuellem Autorenwerkzeug zur Realisierung von 3D-Anwendungen entwickelt. Die neuartigen Erweiterungsansätze zur deklarativen Beschreibung von Raumklang und zur Verhaltens- und Interaktionsmodellierung lassen sich zudem auch unabhängig von der Komponentenarchitektur einsetzen.

Aktuelle Arbeiten zur Realisierung verschiedener 3D-Adaptionsmechanismen für Nutzerpräferenzen, Zielplattformen und Endgeräte sowie zur Ausdehnung der Klassifikations- und Spezifikationskonzepte auf Mixed Reality-Anwendungen runden die vorgestellte Dissertation ab.

Literatur

- [1] R. Dachselt: Eine deklarative Komponentenarchitektur und Interaktionsbausteine für dreidimensionale multimediale Anwendungen. Der Andere Verlag, Tönning, ISBN 3-89959-271-9, Dezember 2004.



- [2] Survey, Taxonomy and Specification of 3D Widgets, 3D Interaction Techniques and 3D Menus. URL: <http://www.3d-components.org/> [Zugriff am 13.10.2006].
- [3] R. Dachsel, A. Hübner: Three-dimensional Menus: A Survey and Taxonomy. Will appear in: Computers & Graphics, International Journal of Systems & Applications in Computer Graphics, Vol. 31/1, 2007.
- [4] H. Hoffmann, R. Dachsel, K. Meißner: An Independent Declarative 3D Audio Format on the Basis of XML. In: Proc. of the 9th Int'l Conf. on Auditory Display: Boston University Publications, pp. 99–102, July 2003.

- [5] R. Dachsel, E. Rukzio: Behavior3D: An XML-Based Framework for 3D Graphics Behavior. In: Proc. of the 8th Int'l Conf. on 3D Web Technology: ACM Press, New York, pp. 101–112, March 2003.



Dr.-Ing. Raimund Dachsel studierte Informatik an der Technischen Universität

Dresden und der University of Glasgow. Nach einem postgradualen Ergänzungsstudium Designinformatik an der Burg Giebichenstein Hochschule für Kunst und Design Halle war er Stipendiat eines DFG-Graduiertenkollegs. Seit der Promotion Ende 2004 ist er als Postdoktorand am Lehrstuhl für Multimedialechnik der TU Dresden tätig. Neben 3D-Grafik liegt der Forschungsfokus zunehmend auf interaktiver Informationsvisualisierung und Benutzungsschnittstellen für Personal Information & Media Management.

Adresse: TU Dresden, Fakultät Informatik, Lehrstuhl Multimedialechnik, D-01062 Dresden, Tel.: +49-351-46338514, Fax: +49-351-46338518, E-Mail: dachsel@acm.org