
**Entwicklung eines dreidimensionalen
Hypermedia-Systems zur Verwaltung und
Synthese historischer Quellen am Beispiel des
Zwangsarbeiterlagers
in Berlin Niederschöneweide**

Diplomarbeit

zur Erlangung des akademischen Grades
Diplom-Informatiker

an der Fachhochschule für
Technik und Wirtschaft in Berlin

Fachbereich Wirtschaftswissenschaften II
Studiengang Angewandte Informatik

1. Betreuer: Prof. Dr. Jung
2. Betreuer: Dipl.-Pol. Layer-Jung

Eingereicht von Chee Fai Cheung
Matrikelnummer: 76900234194
Berlin, den 03.12.2003

Inhaltsverzeichnis

1	Einleitung	- 5 -
2	Aufgabenstellung	- 8 -
3	Anwendungen von 3D-Technologien im historischen Kontext.....	- 9 -
3.1	THEATRON.....	- 9 -
3.2	VRND	- 12 -
3.3	BBC Virtual Tours.....	- 13 -
3.4	Fazit	- 17 -
4	Technologien für die 3D Darstellung im Internet	- 18 -
4.1	Grundlagen Virtual Reality	- 18 -
4.2	Shockwave 3D.....	- 20 -
4.2.1	Überblick.....	- 20 -
4.2.2	Entwicklungssoftware	- 21 -
4.2.3	Betrachtersoftware	- 22 -
4.2.4	Vor- und Nachteile.....	- 22 -
4.3	3D-Engine.....	- 22 -
4.3.1	Überblick.....	- 22 -
4.3.2	Entwicklungswerkzeug	- 23 -
4.3.3	Betrachtersoftware	- 24 -
4.3.4	Vor- und Nachteile.....	- 24 -
4.4	Java 3D	- 25 -
4.4.1	Überblick.....	- 25 -
4.4.2	Entwicklungssoftware	- 27 -
4.4.3	Betrachtersoftware	- 27 -
4.4.4	Vor- und Nachteile.....	- 27 -

4.5	VRML.....	- 28 -
4.5.1	Überblick.....	- 28 -
4.5.1.1	VRML 1.0	- 29 -
4.5.1.2	VRML 2.0	- 29 -
4.5.1.3	VRML 97	- 31 -
4.5.2	Entwicklungssoftware.....	- 31 -
4.5.3	Betrachtersoftware	- 33 -
4.5.4	Vor- und Nachteile.....	- 33 -
4.6	Extensible 3D	- 34 -
4.6.1	Überblick.....	- 34 -
4.6.2	Entwicklungssoftware.....	- 35 -
4.6.3	Betrachtersoftware	- 37 -
4.6.4	Vor- und Nachteile.....	- 37 -
5	Entwicklung des Systems.....	- 38 -
5.1	Analysephase	- 38 -
5.1.1	Ausgangssituation	- 38 -
5.1.2	Zielsetzung	- 39 -
5.1.3	Anforderungen.....	- 39 -
5.1.4	Machbarkeit.....	- 40 -
5.2	Export von 3D Studio Max nach VRML.....	- 42 -
5.3	Optimierung der VRML Datei.....	- 45 -
5.4	Datenbank	- 46 -
5.4.1	Datenbankentwurf	- 46 -
5.4.2	Entity Relationship Modell	- 47 -
5.4.3	Physisches Modell.....	- 48 -
5.5	Schnittstelle zwischen Datenbank und VRML.....	- 50 -
5.5.1	Verbindung zur Datenbank mit PHP.....	- 50 -
5.5.2	Positionswerte der 3D-Modelle.....	- 52 -
5.5.3	Generierung einer VRML-Datei mit PHP.....	- 53 -
5.6	Informationspunkt.....	- 55 -
5.6.1	Viewpoint als Infopunkt.....	- 55 -
5.6.2	Einfügen eines Infopunkts.....	- 57 -

5.7 Bewegen der 3D-Objekte.....	- 59 -
5.7.1 Auswahlverfahren des 3D-Objekts	- 59 -
5.7.2 Steuerkonsole	- 60 -
5.8 Webinterface zur Verwaltung des Systems	- 62 -
6. Ergebnisse	- 64 -
6.1 Bedienung des Prototyps	- 64 -
6.2 Performance Test	- 71 -
7 Ausblick.....	- 73 -

Anhang

Abbildungsverzeichnis	- 75 -
Tabellenverzeichnis	- 77 -
Quellenverzeichnis.....	- 78 -
Glossar	- 83 -
Danksagung.....	- 85 -
Eigenständigkeitserklärung.....	- 86 -

1 Einleitung

„ Ich höre etwas, ich vergesse es, ich sehe etwas, ich erinnere mich daran, ich erlebe etwas, ich verstehe es. “

Chinesisches Sprichwort [1]

Dieses chinesische Sprichwort beschreibt einen der Gründe, warum ein großes Interesse an der dritten Dimension im Internet besteht. In vielen Bereichen lässt sich die Technologie der dreidimensionalen Welt vorteilhaft einsetzen. Die wichtigsten Bereiche sind zum einen E-Business zum anderen wissenschaftliche Anwendungen.

Beim elektronischen Handel liegt der Vorteil darin, dass der Kunde sich in einem virtuellen Kaufhaus bewegen kann und die angebotenen Produkte als 3D-Modell direkt am Bildschirm in einer beliebigen Ansicht anschauen kann. Darüber hinaus kann der Kunde durch 3D Animationen mehr über die Funktionalitäten des Produktes erfahren.

Im Bereich der Immobilie bzw. Architektur kann der Einsatz von 3D-Technologien dazu genutzt werden, um potentiellen Kaufinteressenten vorab einen virtuellen Rundgang durch eine Wohnung, ein Einfamilienhauses oder einer Wohnanlage zu präsentieren. Dadurch können unnötige lange Fahrten gespart werden.

Auf dem Gebiet der Wissenschaft ermöglichen 3D-Technologien komplexe Sachverhalte so verständlich wie möglich darzustellen. Im Medizinbereich können komplexe Operationen am Computer simuliert werden, virtuelle Modellen können erforscht werden.

In der Geschichtswissenschaft eröffnet die 3D-Technologie neue Möglichkeiten, kulturelles Erbe zu erhalten. Architektonische Werke, die noch bestehen oder längst zerstört sind, können als dreidimensionale Modelle rekonstruiert werden. Sie erlauben es, virtuelle Repräsentationen zu betreten und zu erforschen und machen die Geschichte lebendig. Konventionelle Medien wie Geschichtsbücher bieten nur statische 2D Ansichten an. Mit der neuen Technologie kann der Historiker selbst definieren, aus welcher Sicht er das Gebäude betrachten möchte.

Das Ziel dieser Arbeit ist es, für die Berliner Geschichtswerkstatt ein dreidimensionales Modell vom Zwangsarbeiterlager in Berlin Niederschöneweide zu erstellen. Des Weiteren soll ein System entwickelt werden, mit dem Dokumente verwaltet werden, die später in das dreidimensionale Modell integriert werden.

Die Berliner Geschichtswerkstatt [15] ist ein gemeinnützlicher Verein, der im Jahre 1981 gegründet wurde. Seit 1994 beschäftigt sich eine Projektgruppe mit dem Thema NS-Zwangsarbeit in Berlin und Brandenburg. Ziel des Projektes ist die Erforschung der Geschichte des letzten Berliner Zwangsarbeiterlagers. Des Weiteren setzt es sich für den Erhalt der leer stehenden Baracken ein um es als „Informationszentrum zur NS Zwangsarbeit“ zu nutzen.

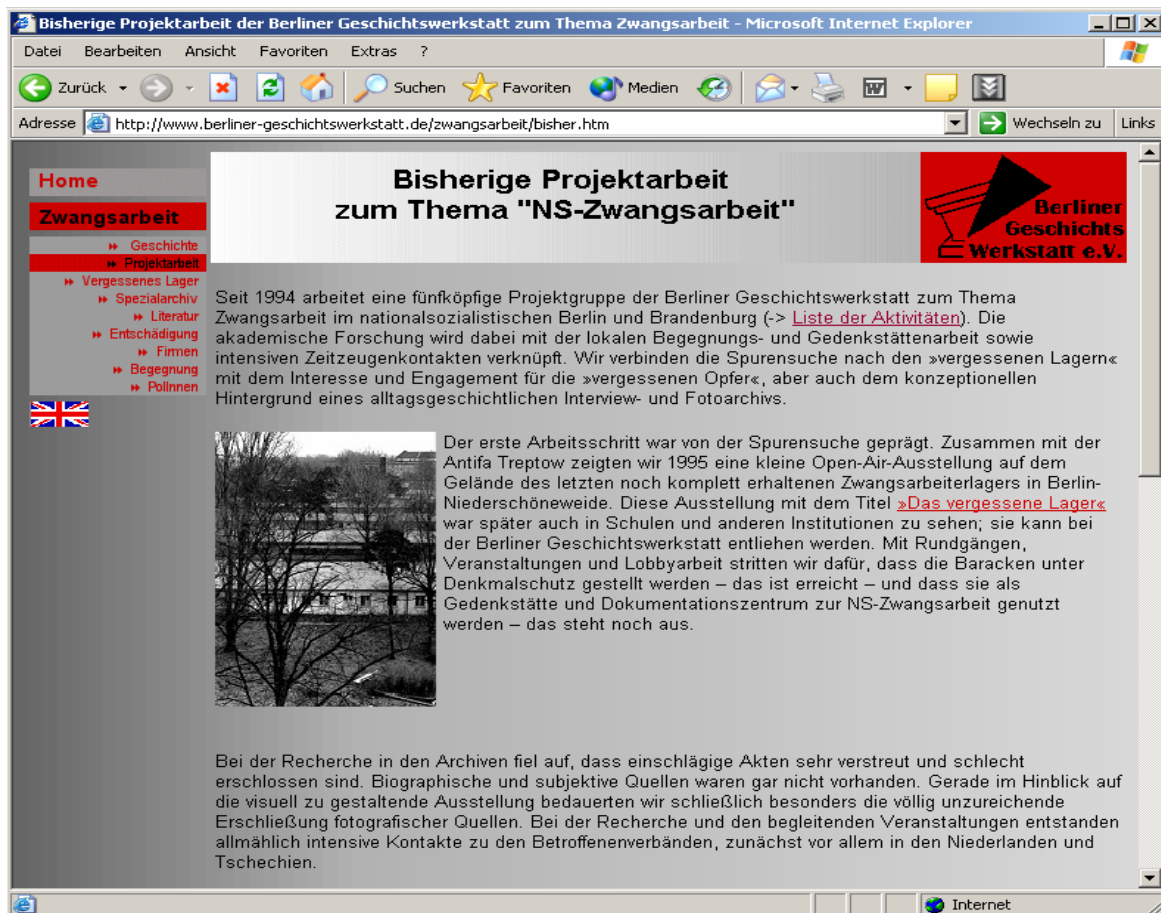


Abbildung 1.1: Homepage der Berliner Geschichtswerkstatt

Im Anschluss an dieses Kapitel werden die einzelnen Aufgaben beschrieben, die für diese Arbeit notwendig sind. Das 3.Kapitel stellt einige Beispiele vor, wie mit heutigen 3D-Technologien historische Gebäude im Internet dargestellt werden können. Für die dreidimensionale Darstellung im WWW gibt es verschiedene Technologien. Einige Technologien werden im 4. Kapitel vorgestellt. Das 5.Kapitel befasst sich mit der Entwicklung des Prototyps. Es beginnt mit einer Analysephase, später werden Konzepte zur Realisierung vorgestellt. Kapitel 6 stellt den Prototypen vor. Im abschließenden Kapitel gibt es eine Zusammenfassung und es werden mögliche Erweiterungen diskutiert.

2 Aufgabenstellung

Die Diplomarbeit befasst sich mit der Entwicklung eines 3D-Hypermediasystems zur Verwaltung und Synthese historischer Quellen, das für die Berliner Geschichtswerkstatt entwickelt wird. Als Beispiel dient das Zwangsarbeiterlager in Berlin Niederschöneweide.

Die Diplomarbeit umfasst drei virtuelle Welten. In jeder Welt kann sich der Benutzer frei bewegen und die Objekte aus beliebigen Blickwinkeln und Entfernungen betrachten. Zu jeder 3D-Rekonstruktion gibt es eine ausführliche Dokumentation in Form von Bildern und Texten. Zur Verwaltung der Dokumente wird eine Datenbank eingesetzt.

Im Zusammenhang mit der Erstellung des Prototyps lassen sich folgende Teilaufgaben identifizieren:

Untersuchung der gängigsten Technologien für die 3D-Darstellung unter dem Aspekt der Interaktionsmöglichkeit, Anbindung an eine Datenbank, Entwicklungssoftware und Betrachtersoftware

Exportieren der vorhandenen 3D-Modelle in ein zuvor gewähltes 3D-Format mit anschließender Optimierung der Modelle

Entwicklung einer Datenbankstruktur, womit 3D-Objekte und historische Daten verwaltet werden können einschließlich einer grafischen Benutzeroberfläche.

Ausarbeiten einer Methode, um historische Daten in Bezug auf das jeweilige 3D-Objekt zu visualisieren.

Entwicklung einer Möglichkeit mit der man nachträglich die Position von 3D-Objekten interaktiv verändern kann.

3 Anwendungen von 3D-Technologien im historischen Kontext

In diesem Kapitel werden einige Beispiele vorgestellt, die ich bei der Recherche im Internet gefunden habe. Am Ende gibt noch eine Zusammenfassung der Beispiele.

3.1 THEATRON

Auf der Internetseite von THEATRON [16] findet man ein Informationssystem, das sich speziell mit der Rekonstruktion von Theatergebäuden im alten Europa beschäftigt. Das Projekt wurde in Zusammenhang mit der Europäischen Kommission sowie Vertretern aus akademischen Instituten, Spezialisten aus dem Bereich der Informationstechnologie und Architekten erstellt. Das System wurde mit Hilfe von VRML und HTML realisiert. Um das Informationssystem zu nutzen, muss ein Programm auf dem lokalen PC installiert werden. Ruft man das Programm auf, so wird eine Verbindung über das Internet zum Webserver von Theatron aufgebaut. Über diese Verbindung werden Daten vom Server geladen und auf dem Client dargestellt. Für die Anzeige wird ein Internetbrowser vorzugsweise der Internet Explorer 5 oder eine höhere Version benötigt. Die Startseite teilt sich in drei Frames auf. Die Menüs befinden sich am oberen und unteren Bildschirmrand. Der mittlere Frame ist für die Ausgabe von Text und Grafik zuständig. Um eine bessere Übersicht über die verfügbaren 3D-Objekte zu erhalten, kann der Benutzer die Gebäude nach den Kriterien Zeit, Topologie, geographische Lage oder Namen sortiert angezeigt bekommen (siehe Abbildung 3.1).

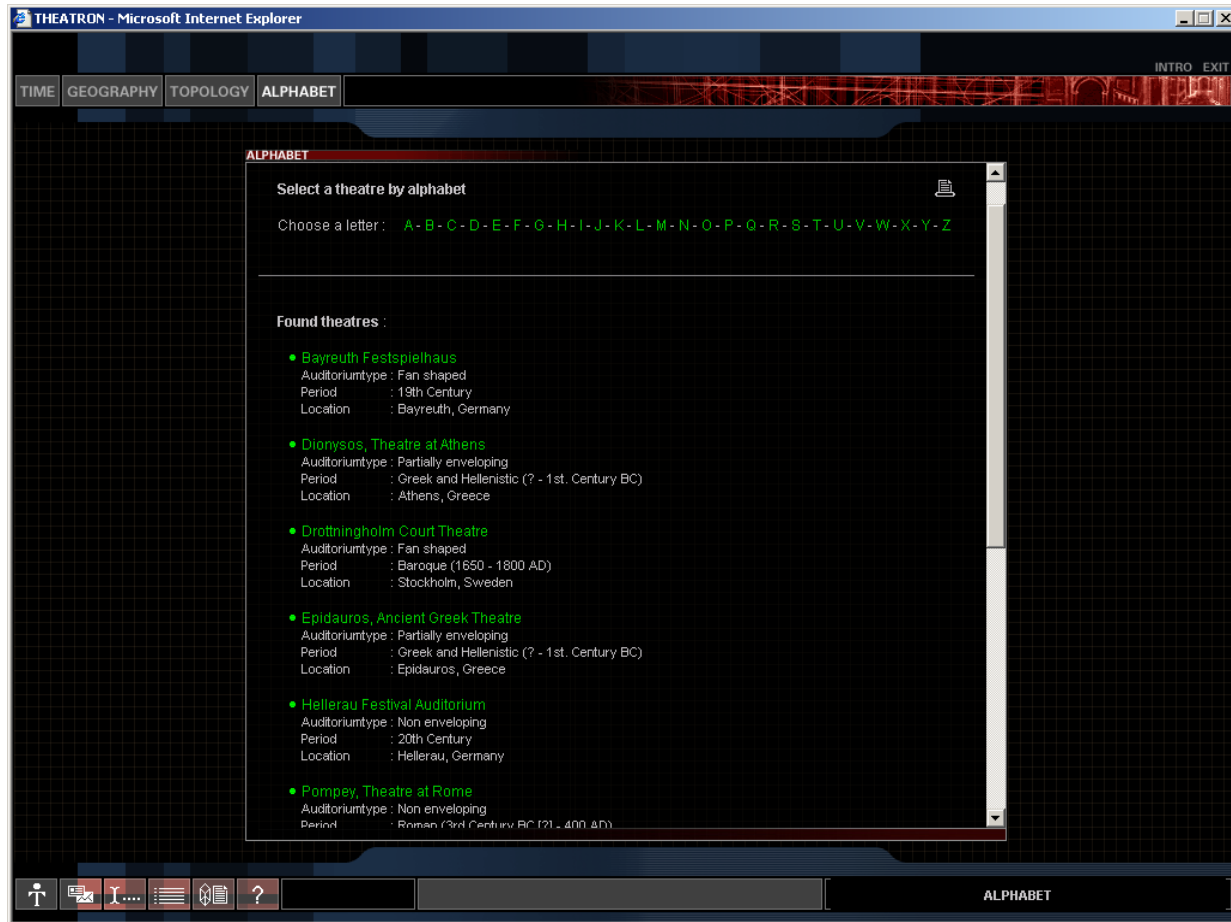


Abbildung 3.1: Menü von Theatron

In der Demoverision ist nur das Objekt „Theater von Dionysos“ verfügbar. Die 3D-Szene wird in den mittleren Frame geladen. Links und rechts der 3D Szene werden in separaten Frames Informationen zum Objekt angezeigt.

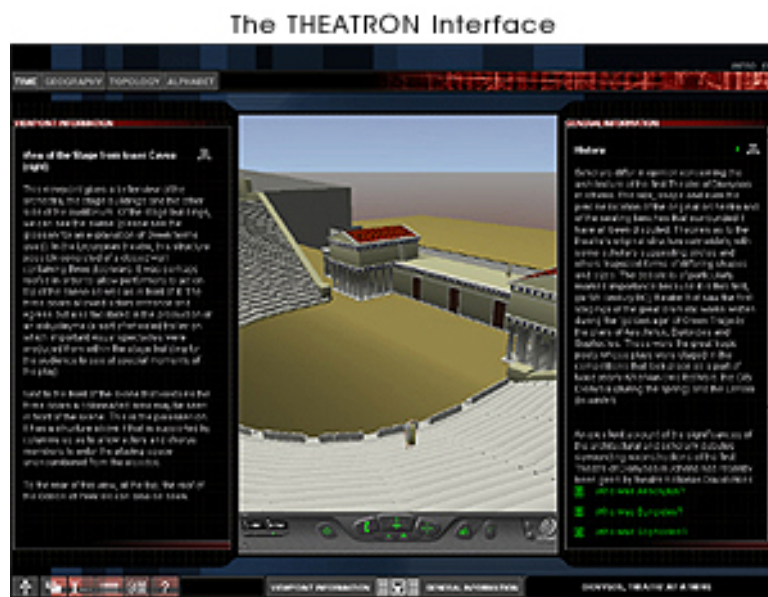


Abbildung 3.2: Interface, Theater von Dionysos

Das Navigieren erfolgt mit der Maus. Der Besuch der vordefinierten Aussichtspunkte gibt dem Nutzer in kürzester Zeit einen guten Überblick über die 3D-Welt. Abhängig vom Standort werden im linken Anzeigefenster Textinformationen aus einer Datenbank angezeigt. Zusätzliche Links führen zu Fotos und Panoramen, die dem Nutzer einen Vergleich zwischen der realen und der virtuellen Welt ermöglicht.

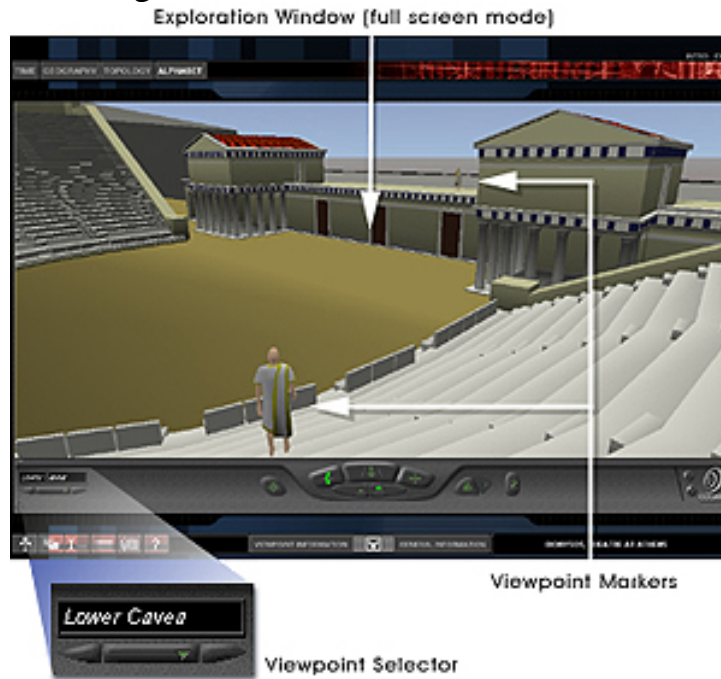


Abbildung 3.3: Vollsicht auf das Theater von Dionysos

Für eine größere Darstellung der 3D-Szene können die Frames ausgeblendet werden (Abb. 3.3).

Bei der Benutzung des Systems sollten folgende Punkte beachtet werden. Eine CPU mit mindestens 400 MHz und 128 MB Hauptspeicher sowie eine Soundkarte und eine Internetverbindung sollten verfügbar sein. Des Weiteren benötigt man ein VRML-Plugin, den Quicktime Movieplayer, ein Macromedia Flash Plugin und den Webbrowser Internet Explorer ab der Version 5.

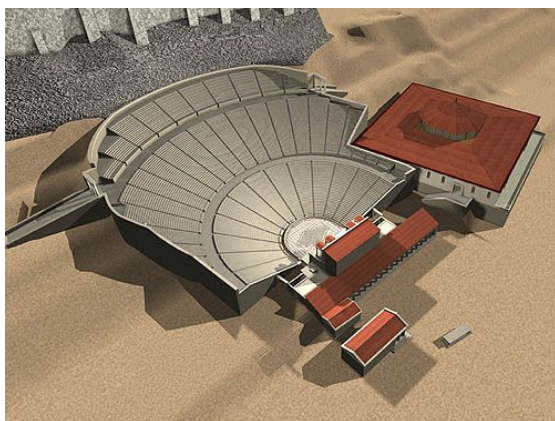


Abbildung 3.4: Sicht von Oben



Abbildung 3.5: Sicht von der Seite

Ingesamt ist das System Theatron gut gelungen. Durch das einfache Benutzerinterface kann man in kürzester Zeit mit dem System umgehen. Der Benutzer erhält detaillierte Informationen zu den Objekten in Form von Texten und Bildern. Zur Zeit sind 10 Gebäude im System verfügbar. Um alle 3D-Modelle zu betrachten, muss eine Nutzungsgebühr gezahlt werden. Der Nachteil des Systems besteht darin, dass man permanent mit dem Server verbunden sein muss. Dabei entstehen hohe Onlinekosten.

3.2 VRND

Auf der Internetseite von VRND [17] findet man ein Referenzbeispiel, wie man mit Hilfe einer „3D-Engine“ historische Gebäude in einer virtuellen Welt rekonstruieren kann. Die Projektgruppe besteht aus Mitarbeitern aus den Gebieten der Computergrafik, Programmierung und VR- und Echtzeit-Visualisierung. Für die Erzeugung der 3D-Welt benutzen sie die Visualisierungstechnik aus der Spielebranche. Computerspiele wie Quake [18], Unreal [19] oder Halflife [20] nutzen eine so genannte „3D-Engine“, um eine realistische 3D-Umgebung zu erzeugen. Was eine „3D-Engine“ genau ist, wird im 4. Kapitel erläutert.

In diesem Beispiel kann man die 800 Jahre alte Kathedrale von Notré Dame in Echtzeit erkunden. Um in die 3D Welt einzutauchen, muss ein Programm installiert werden. Dieses Programm beinhaltet die ganze 3D-Szene. Das hat den Vorteil, dass der User beim Betrachten der 3D-Welt nicht permanent mit dem Internet verbunden sein muss. Auf Wunsch kann eine Verbindung zum Server aufgebaut werden, um Multiuserfunktionen wie den Chat zu benutzen. Die Navigation erfolgt sowohl durch die Maus als auch mit der Tastatur. Das besondere an diesem System ist, dass der Nutzer eine Taschenlampe aktivieren kann, die in den dunklen Gängen der Kathedrale ganz hilfreich ist. Auch die Funktion des Fotografierens wurde integriert. Die geschossenen Bilder werden in einem Ordner abgelegt und können später betrachtet werden.

Auffallend ist, dass man in der virtuellen Kathedrale keine Informationen zu den Objekten erhält. Die Version, die man auf der Homepage angeboten bekommt, ist im Betastadium. Eine Fortführung des Projektes ist unwahrscheinlich, da seit 1999 keine neue Version und keine weiteren Informationen erhältlich sind.

Um eine optimale Darstellung zu erhalten wird eine 3D-Beschleunigerkarte benötigt. Des Weiteren sollte ein Pentium3 400 MHz und 128 MB Hauptspeicher sowie eine Soundkarte und eine ISDN Verbindung für den Multiuserbetrieb vorhanden sein.



Abbildung 3.6: Screenshots von dem System VRND

3.3 BBC Virtual Tours

BBC ist eine Rundfunkanstalt in Großbritannien und wurde 1922 von John Reith gegründet. Auf der Homepage der British Broadcast Corporation [21] gibt es eine Rubrik, die sich mit der Geschichte Großbritanniens beschäftigt. Dort können architektonische Bauwerke und historische Objekte als begehbare 3D-Welten erkundet werden. Auch Panoramabilder von historischen Stätten sind verfügbar.



Abbildung 3.7: BBC Virtual Tours

Man kann z.B. das alte Segelschiff von Kapitain Cook oder die mittelalterliche St.Edmunds Abtei erkunden. Die Navigation kann über die Maus oder über die Tastatur erfolgen. Da die 3D-Welten mit VRML erstellt worden sind, wird hierfür ein VRML-Plugin für den Webbrowser benötigt. Leider bietet dieses Informationssystem keine abrufbaren Informationen innerhalb der 3D Welt an, wie es bei Theatron der Fall ist. Information zu den Objekten werden nur vor dem Betreten der 3D Welt in Form von Texten und Screenshots angeboten. Für zusätzliche Informationen werden Links zu anderen Internetseiten angeboten.

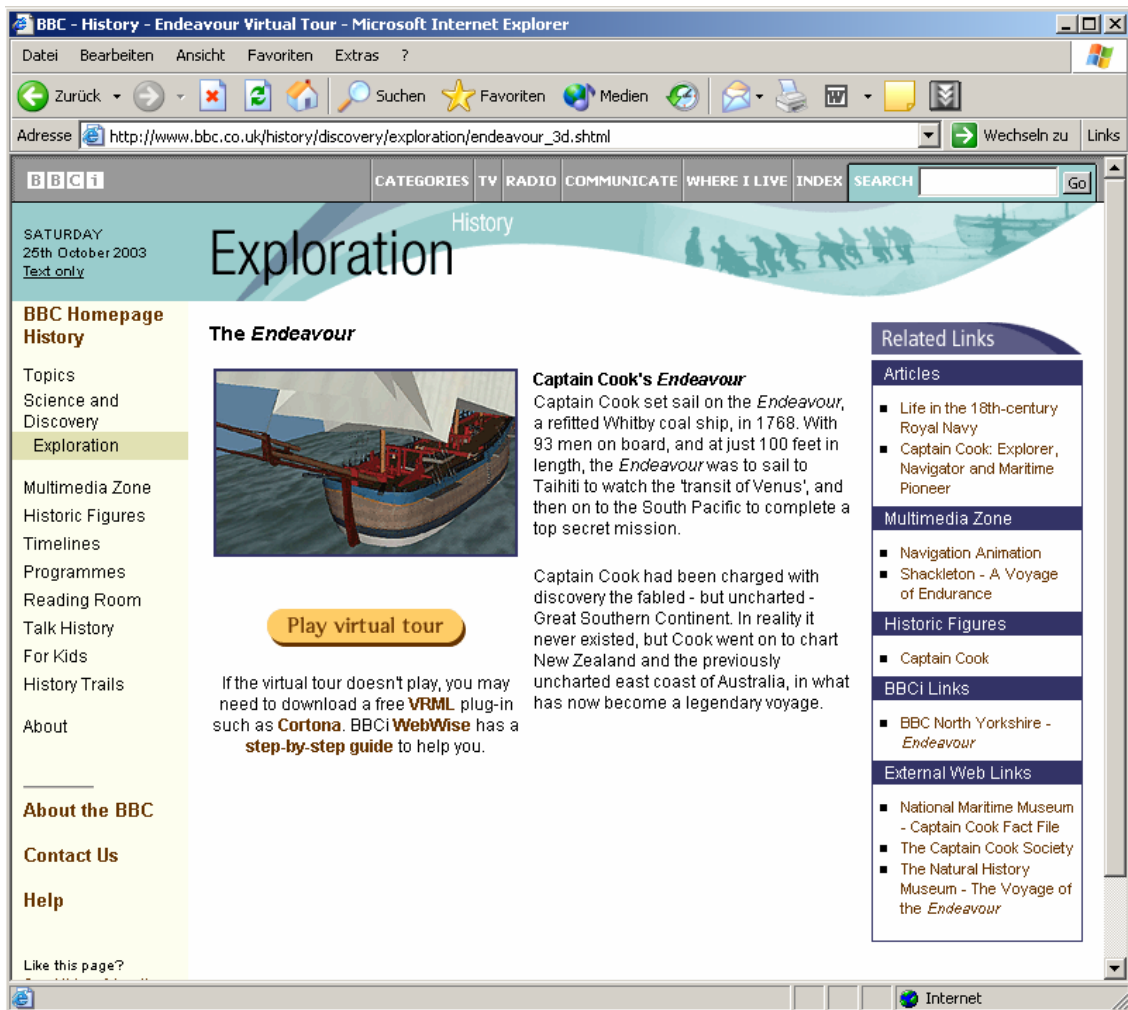


Abbildung 3.8: Einstiegspunkt Endeavour

In Abbildung 3.8 sieht man den Einstiegspunkt zum Segelschiff Endeavour. In der Mitte des Bildschirms gibt es eine Kurzfassung zur Geschichte des Schiffes. Für ausführlichere Informationen, sind auf der rechten Seite Links aufgeführt. Die gleiche Struktur sieht man auch in den Abbildungen der St.Edmunds Abtei.

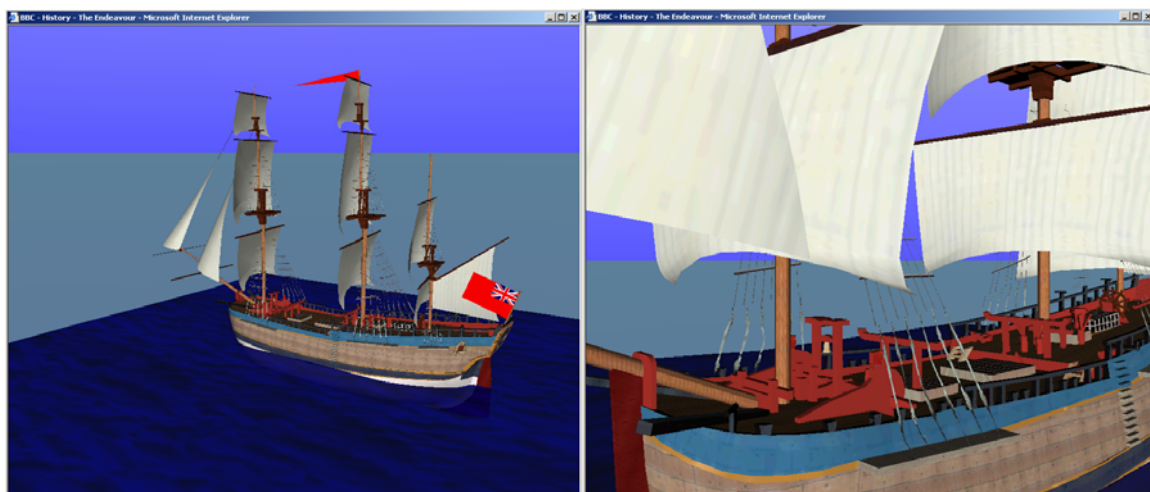


Abbildung 3.9: VRML Modell des Segelschiffs Endeavour

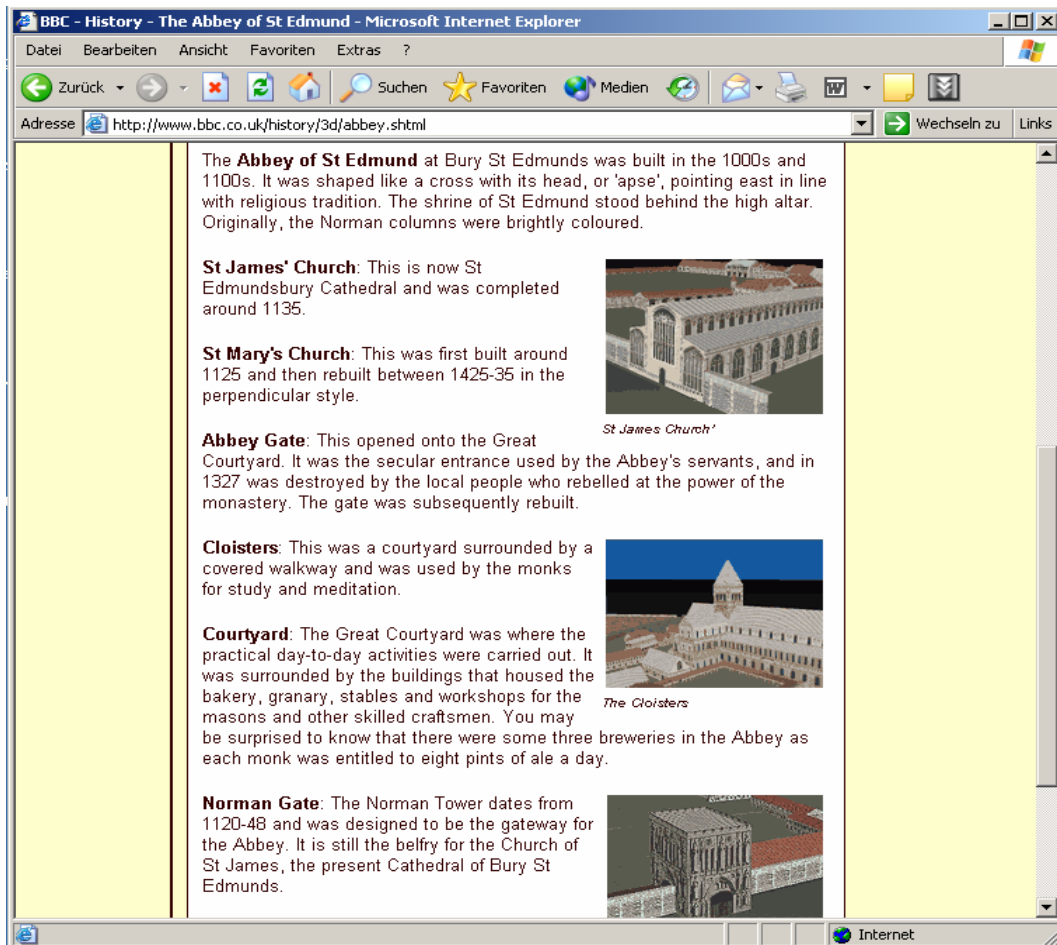


Abbildung 3.10: Einstiegspunkt St.Edmunds Abtei

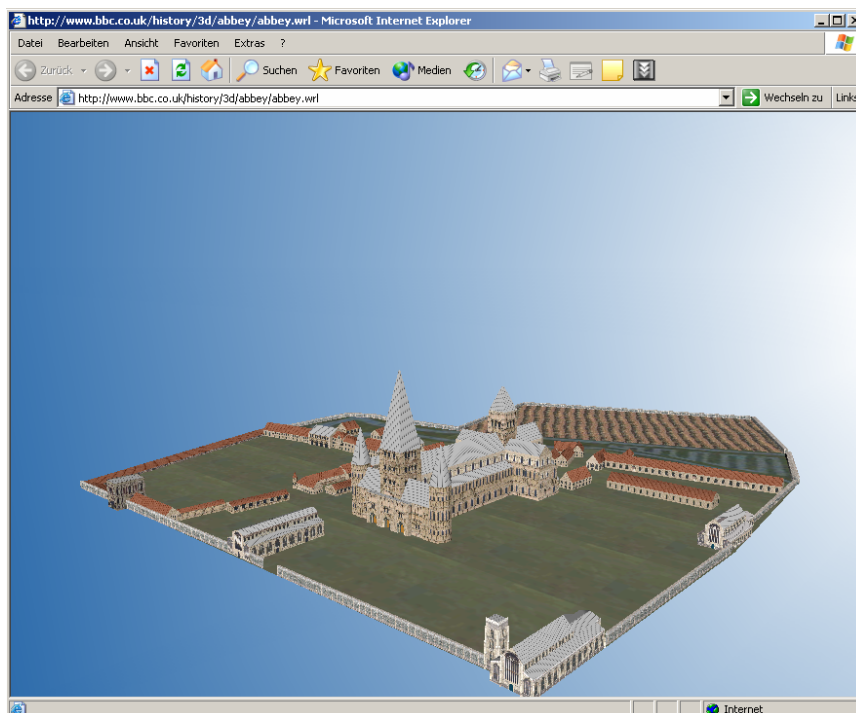


Abbildung 3.11: VRML Modell der St.Edmunds Abtei

3.4 Fazit

Theatron richtet sich an Personen, die sich mit der Materie intensiv beschäftigen, da der Informationsgehalt hoch ist. Ein Nutzungsentgelt gibt einen weiteren Indiz dafür, dass es sich an Personen richtet, die sich ernsthaft mit der Geschichte der Theater auseinandersetzen. Die Darstellung der Informationen ist übersichtlich gestaltet. Die VRML-Modelle sind gut modelliert und brauchen keine langen Ladezeiten. Der Nachteil dieses Systems ist, dass man ständig mit dem Server verbunden sein muss. Außerdem müssen eine Vielzahl von Plugins installiert werden.

VRND hat dagegen keine konkrete Zielgruppe. Historische Informationen in der 3D-Welt sucht man vergebens. Das Primärziel des Projekts war, eine realistisch wirkende Abbildung der Kathedrale in Echtzeit darzustellen. Da VRML die Qualitätsanforderungen nicht erfüllte, setzte man eine 3D-Engine ein. Vergleicht man die 3D-Welt mit den anderen Beispielen, so erkennt man, dass die Kathedrale mit der 3D-Engine weitaus realistischer wirkt. Ein weiterer Vorteil bei diesem System ist, dass es keine zusätzlichen Plugins benötigt und dass man nicht permanent mit dem Server verbunden sein muss. Meiner Meinung nach bietet das System durchaus eine gelungene Alternative im Bereich der 3D-Visualisierung.

Nach dem Informationsgehalt ist das System von BBC Virtual Tours sowohl für Historiker als auch für neugierige Besucher gedacht. Die Dokumentationen sind deshalb kompakt und übersichtlich gehalten. Möchte man sich ausgiebiger mit dem Thema beschäftigen, leiten zusätzliche Links den Benutzer zu weiteren Informationsquellen. Die Modelle sind gut modelliert und brauchen nur kurze Ladezeiten. Auch für dieses System muss ein Plugin installiert werden.

System	Zielgruppe	Technologie	Informationsgehalt
Theatron	Historiker	VRML,HTML	hoch
VRND	-	3D-Game Engine	-
BBC Virtual Tours	Historiker, Laie	VRML,HTML	mittel bis hoch

Tabelle 3.1: Systemvergleich

Die Recherche hat gezeigt, dass im Internet nicht viele 3D-Rekonstruktionen von historischen Gebäuden existieren. Auffallend ist, dass bis auf das System von Theatron niemand eine Integration der Informationen in die 3D Welt vorgenommen hat. Durch die Synthese der Dokumentation mit einer virtuellen Welt kann die Effizienz eines solchen Informationssystems gesteigert werden. Der Nutzer kann die Information besser zuordnen und verarbeiten, wenn die Dokumentation im Zusammenhang mit der virtuellen Welt erscheint.

4 Technologien für die 3D Darstellung im Internet

4.1 Grundlagen Virtual Reality

Virtual Reality ist eine Technologie, die es Anwendern erlaubt, eine dreidimensionale vom Computer generierte Welt zu erleben. Der Nutzer taucht in die virtuelle Welt ein und kann mit ihr interagieren. Die vom Computer generierte Welt kann eine Abbildung der realen Welt oder auch eine fiktive Welt sein. Da noch keine 3D-Ausgabemedien existieren, muss 3D-Visualisierungssoftware Methoden der Computergrafik nutzen um räumliche Objekte auf flachen Ausgabemedien zu simulieren. Um die räumliche Illusion zu erzeugen, hat man verschiedene Systeme und Geräte entwickelt.

Das einfachste Virtual-Reality-System ist die Desktop VR, auch Window on World genannt, da der Benutzer von einem außerhalb gelegenen Blickpunkts, wie durch ein Fenster in die Virtuelle Welt hineinschaut. Die virtuelle Welt wird auf einem flachen Computermonitor dargestellt, für die Interaktion wird entweder eine Computermaus oder die Tastatur benutzt.

Im Gegensatz zum Desktop VR trägt der Anwender mit dem Head Mounted Display (HMD) die gesamte Visualisierungseinrichtung in Form eines Helmes auf dem Kopf. Dabei werden durch den Helm zwei kleine CRT- oder LCD-Bildschirme in geringem Abstand vor den Augen positioniert. Durch eine spezielle Optik wird das Sichtfeld vergrößert. Um dem Betrachter das Gefühl zu geben, vollständig in die Simulation eingetaucht zu sein, wird bei HMDs ein Stereolautsprecher als Kopfhörer in den Helm integriert und ein Trackingsystem eingebaut. Das Trackingsystem besteht aus einer Vielzahl von Sensoren, die die Kopfbewegungen des Nutzers registrieren und so die virtuelle Welt dem jeweiligen Blickwinkel anpassen.



Abbildung 4.1: HMD

Ein CAVE Automatic Virtual Environment (kurz CAVE) ist eine Simulationseinrichtung, die aus einem kleinen Raum besteht, auf dessen transparente Wände mittels Rückwandprojektion drei oder mehr Großbildprojektoren Bilder projizieren. Die Simulationsteilnehmer befinden sich im Inneren des Raumes und nehmen die virtuelle Umgebung an den Wänden durch spezielle 3D-Brillen wahr. In den modernen CAVEs wird der Boden des Raumes als zusätzliches Display verwendet, auf den das Bild von oben projiziert wird. Durch mehrere Lautsprecher wird ein 3D-Sound erzeugt, der die Wahrnehmung des Nutzers in einer virtuellen Welt verstärkt.

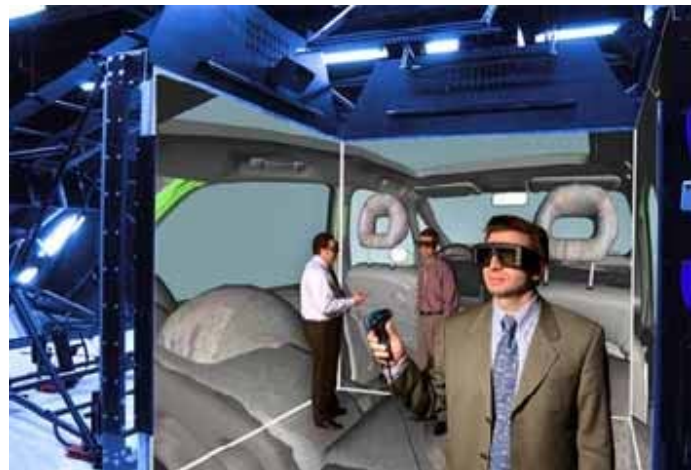
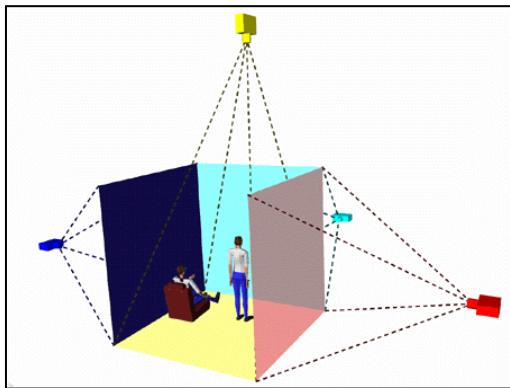


Abbildung 4.2: CAVE

HMD-Systeme werden meist in Kombination mit einem Datenhandschuh genutzt. Alle Handbewegungen, die der Nutzer in der realen Welt macht, werden über Sensoren aufgenommen und in die virtuelle Welt übertragen. Diese können zur Navigation oder zur Steuerung einer virtuellen Hand genutzt werden. Der Nutzer kann über den Datenhandschuh Objekte in der virtuellen Welt greifen und ablegen oder durch Schalter betätigen. Dadurch fühlt sich der Nutzer stärker in die virtuelle Welt integriert.



Abbildung 4.3: Datenhandschuh

4.2 Shockwave 3D

Die Schlüsselkomponente für das zu entwickelnde System ist die 3D Visualisierungstechnologie. Daher werden im Folgenden die einzelnen 3D-Technologien vorgestellt.

4.2.1 Überblick

Shockwave3D wurde von Macromedia [22] in Zusammenarbeit mit den Intel Architecture Labs (IAL) für die Director Version 8.5 entwickelt. Die Firma Intel hat die 3D-Technologie entwickelt und Macromedia sich mit der Integration in das Produkt beschäftigt.

Macromedia Director ist ein Programm mit dem man multimediale Anwendungen erstellen kann. Da diese Anwendungen große Datenmengen enthalten, wurden sie auf CD basierenden Medien publiziert. Durch die Einführung der Breitbandübertragungstechnik kann man heute mit einer wesentlich höheren Geschwindigkeit im Internet surfen. Deshalb führte Macromedia die Shockwave Technologie für Director ein, die ab Version Director 8.5 mit 3D-Eigenschaften erweitert wurde.

Mit Shockwave werden multimediale Anwendungen durch Einsatz von optimierten Komprimierungsalgorithmen um ein vielfaches verkleinert, um so lange Ladenzeiten im Internet zu vermeiden.

Einsteiger können, wenn sie die eingesetzte "Film"-Metapher einmal verstanden haben, relativ einfach Multimedia-Präsentationen erstellen. Fortgeschrittene Benutzer können mit der integrierten Skriptsprache "Lingo" komplexe Simulations- und Interaktionsumgebungen programmieren. Auch besteht die Möglichkeit komplette Szenen aus 3D-Modellierprogrammen wie 3D Studio Max oder Cinema 4D in Director zu importieren.

Wenn neue spezialisierte Funktionen notwendig sind, lassen sich zusätzliche Programmmodule, sogenannte „Xtras“, in Director einbinden. Durch diese Eigenschaft kann Director erweitert werden. Xtras werden in der Regel von Drittanbietern angeboten und verlangen meist für die Nutzung eine Lizenzgebühr. Zum Beispiel bietet die Firma Sight & Sound Software [23] das Xtra „Datagrip Net“ an, mit Hilfe dessen man Datenbanken über ein Netzwerk benutzen kann.

4.2.2 Entwicklungssoftware

Basis einer Director-Anwendung ist der "Film" mit einer Bühne, seinen Darstellern und dem Drehbuch. Mit der integrierten Programmiersprache "Lingo" kann man Anwendungen ein hohes Maß an Interaktivität geben. In der Abbildung 4.4 sehen wir die Entwicklungsumgebung von Director. Auf der linken Seite gibt es Werkzeuge für das Zeichnen einfacher Objekte. In der Mitte ist die Bühne. Darunter sehen wir das Drehbuch und die Darsteller. Im Drehbuch wird festgelegt, welcher Darsteller auf der Bühne mitspielt und zum welchem Zeitpunkt er erscheinen soll. Um Interaktionen zu erzeugen, werden Darsteller meist mit Verhaltenskripten ausgestattet, um auf bestimmte Aktionen zu reagieren. So kann der Darsteller z.B. auf das Klicken der Maus reagieren und als Effekt eine Animation auslösen. Einen Ausschnitt aus solch einem Verhaltenskript sieht man auf der rechten Bildhälfte in Abbildung 4.4.

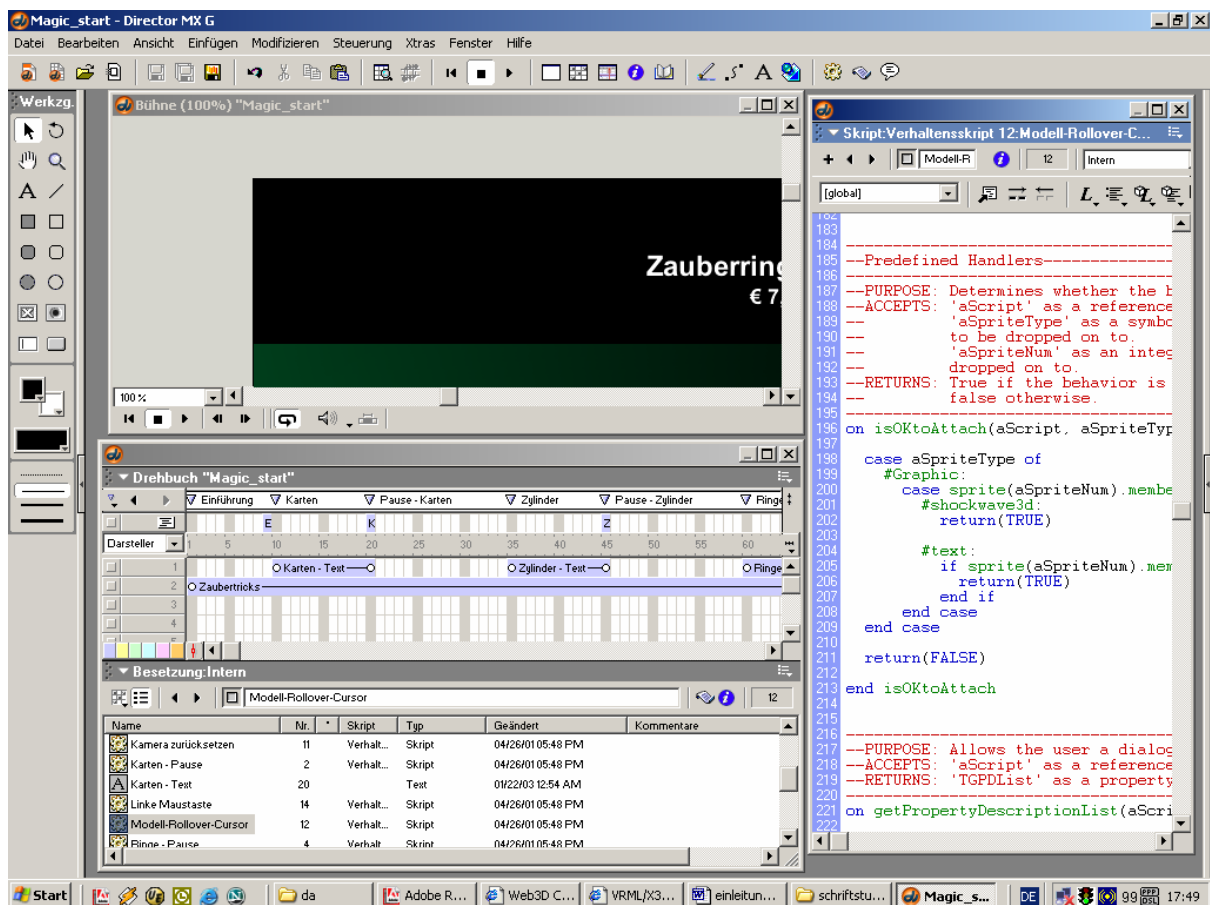


Abbildung 4.4: Benutzeroberfläche von Macromedia Director

4.2.3 Betrachtersoftware

Für das Betrachten von Shockwave 3D-Szenen wird ein Plugin für den Internetbrowser benötigt. Das Plugin kann man bei Macromedia herunterladen, es wird für verschiedene Betriebssysteme und Browser angeboten.

4.2.4 Vor- und Nachteile

Mit Director lassen sich über die grafische Benutzeroberfläche schnell einfache multimediale Anwendungen erstellen. Unter Zuhilfenahme der Skriptsprache Lingo können komplexe Simulations- und Interaktionsumgebungen programmiert werden. Erweiterungsmöglichkeiten werden über Xtras gewährleistet.

Mit der Shockwave-Technologie hat Macromedia zwar eine Möglichkeit geschaffen, multimediale Anwendungen über das Internet zu publizieren, doch lassen sich mit dieser Technik nur kleine 3D-Szenen präsentieren. Komplexe Szenen mit vielen 3D-Objekten brauchen trotz der Komprimierungsalgorithmen zu viel Speicherplatz und führen im Internet zu langen Ladezeiten. Außerdem wird ein Plugin für den Webbrowser benötigt.

Vorteil	Nachteil
Applikation können schnell erstellt werden	Plugin wird benötigt
Animation und Interaktion mit Lingo programmierbar	nicht für große 3D-Welten geeignet
Erweiterungsmöglichkeit durch Xtras	

Tabelle 4.1: Vor- und Nachteile Macromedia Director

4.3 3D-Engine

4.3.1 Überblick

Eine 3D-Engine ist im Grunde eine Programmierbibliothek. In dieser Bibliothek befinden sich die Strukturen, Funktionen und Algorithmen, welche dazu dienen, 3D-Objekte möglichst effizient auf einer 2D-Fläche darzustellen. Funktionen wie Lichtreflexionen an Objekten brauchen nicht mehr selbst programmiert zu werden.

Viele 3D-Engines stellen aber nicht nur die reinen Visualisierungsfunktionen zur Verfügung. So werden Unterschiede zwischen der vielfältigen 3D-Hardware aufgelöst, physikalische Kräfte auf 3D Objekte simuliert oder Kollisionen behandelt. Mit Hilfe solcher Bibliotheken kann man wesentlich einfacher als mit den allgemeinen 3D-APIs – wie z.B. OpenGL oder DirectX - 3D-Welten erstellen. Inwieweit bzw. ob überhaupt bestimmte Elemente der 3D-Programmierung durch Kapselung vereinfacht werden, hängt von der jeweiligen 3D-Engine ab. Hier reicht die Spannweite von der alleinigen Unterstützung von Primitiven bis hin zu komplexen Modellen und/ oder kompletten Welten.

3D-Engines sind keine „Click-and-Play“ Software. Um mit einer 3D-Engine arbeiten zu können, bedarf es guter Programmierkenntnisse sowie Vertrautheit mit der Architektur der 3D-Engine und deren Anwendung.

Durch die kontinuierliche Entwicklung von leistungsfähigen Computern sind aktuelle 3D-Engines in der Lage, realitätsnahe 3D-Welten mit besonderen Stimmungen und Atmosphären zu erzeugen und so beim Betrachter den Eindruck erwecken, sich in dieser Welt zu bewegen und mit ihr zu interagieren. Besonders die Bereiche Entertainment und Simulation sorgen dafür, dass derartiger 3D-Engines weiterentwickelt und ständig verbessert werden um noch realistischere Welten zu erzeugen. Es gibt frei erhältliche und kommerzielle 3D-Engines. Letztere sind für den professionellen Einsatz wie z.B. für PC Spiele gedacht und kosten mehrere hunderttausend US Dollar.

Werden Datenbanken genutzt, können diese über ODBC angesprochen werden. Die Abkürzung ODBC steht für „Open Database Connectivity“ und heißt soviel wie offene Datenbankverbindung. ODBC wurde von der Firma Microsoft entwickelt und stellt eine Programmierschnittstelle dar, die den Zugriff auf unterschiedliche Datenbanken gewährleisten soll.

4.3.2 Entwicklungswerkzeug

Um eine 3D-Welt mit einer 3D-Engine zu erstellen wird eine Programmiersprache benötigt. Führende PC-Spielehersteller wie ID Software [24], Epicgames [25] oder Valve [26] bieten auf ihrer Homepage kostenlos ältere 3D-Engine zur Entwicklung eigener 3DWelten zum Download an. Eine grobe Übersicht von verfügbaren 3D-Engines kann man auf der Internetseite von 3dengines [27] bekommen.

4.3.3 Betrachtersoftware

Für das Betrachten einer virtuellen Welt, die auf einer 3D-Engine basiert, wird keine zusätzliche Betrachtersoftware benötigt.

4.3.4 Vor- und Nachteile

Die Vorteile der Informationsvisualisierung mittels einer 3D-Engine liegen darin, dass die erzeugten 3D-Welten sehr realistisch wirken. Das liegt daran, dass die Engine spezielle Funktionen und Effekte anbietet, um z.B. Lichtreflexionen an 3D-Objekten darzustellen. Der Endbenutzer kann sich in den virtuellen Welten frei bewegen. Interaktionen und Animationen werden mit einer ausgewählten Programmiersprache programmiert und können dadurch selbst definiert werden. Des Weiteren müssen keine zusätzliche Betrachtersoftware installiert werden.

Hauptnachteil ist der enorme Rechenaufwand der bei dieser Art Anwendungen entsteht. Dadurch ist man gezwungen, die Detailtreue so zu reduzieren, dass die Nutzbarkeit noch gewährleistet ist. Gute Programmierkenntnisse sowie Kenntnisse über die Funktionsweise der 3D-Engine sind für die Erstellung von 3D-Welten unabdingbar. Die Dokumentation der einzelnen 3D-Engines ist meist unzureichend. Dadurch ist die Fehlersuche meist schwierig.

Vorteil	Nachteil
3D Welten wirken sehr realistisch	hoher Rechenaufwand
kein Plugin notwendig	schlechte Dokumentation
Animation und Interaktion frei programmierbar	gute Programmierkenntnisse
	gute Kenntnisse über die Funktionsweise der 3D-Engine

Tabelle 4.2: Vor- und Nachteile von 3D-Gameengine

4.4 Java 3D

4.4.1 Überblick

Java 3D ist eine Erweiterung der Programmiersprache Java, die von der Firma Sun [28] entwickelt wurde. Mit der Entwicklung des 3D API's wollte man die Möglichkeit schaffen, 3D-Anwendungen auf einfache Weise zu erstellen, die auf allen Java-fähigen Plattformen laufen sollen. Java3D basiert auf der Java-Version 1.2 und den Low-Level-Treibern (z.B. OpenGL bzw. DirectX).

Um eine virtuelle Welt mit Java3D darzustellen, müssen alle Objekte des Java3D-Universums in einen Szenegraphen eingebunden werden. Die folgende Abbildung zeigt einen Beispielszenegraphen für ein einfaches Universum, welches Teile des Sonnensystems modelliert.

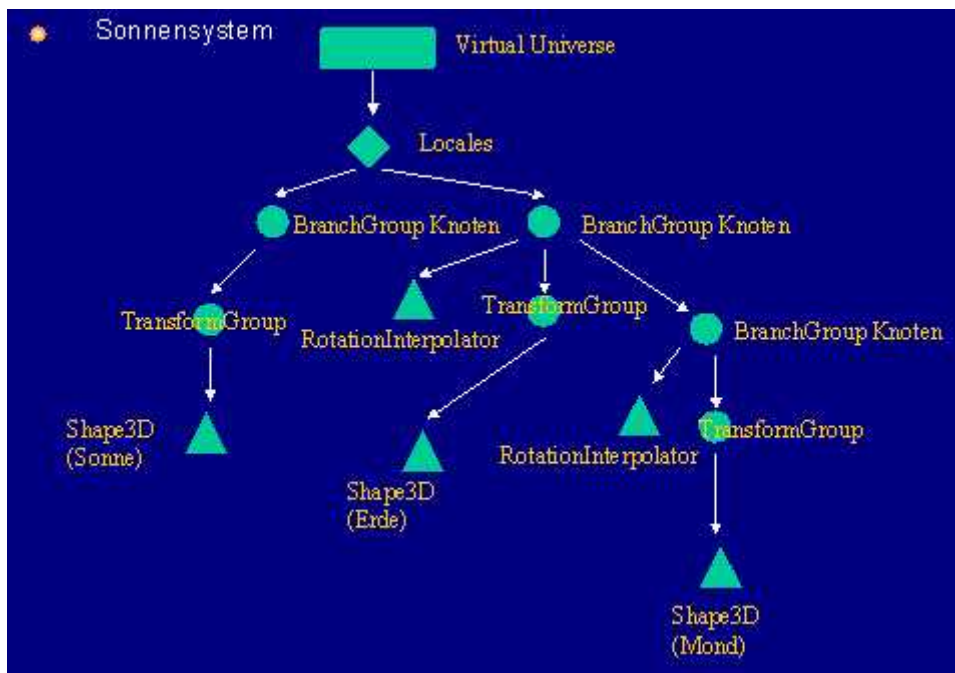


Abbildung 4.5: Java3D Virtual Universe

Quelle des Szenegraphen ist das Virtual Universe Objekt, von der in jeder Anwendung nur eine Instanz existieren darf. Das Virtual Universe verwaltet mehrere Locales, welche verschiedene Lokalitäten in dem virtuellen Universum implementieren.

An die Locales können BranchGroupNodes angehängt werden, an die wiederum beliebig viele weitere Knoten angehängt werden können. Die angehängten Objekte sind in Groups (Gruppen) und Leafs (Blätter) unterteilt, wobei an die Leafs keine weiteren Objekte angehängt werden können. Z.B. sind alle

grafischen Objekte (Kugel, Würfel, Pyramide, ...) eine Ableitung der Leaf Klasse; Animationen werden in einer Gruppe modelliert, da sie sich auch auf mehrere Objekte auswirken können.

Die wichtigsten Leaf-Nodes sind

Shape3D

Die Basisklasse aller konkreten Objekte wie Kugel, Zylinder, Pyramide,

Behaviours

Diese Klassen enthalten Informationen über das Verhalten der zugehörigen Objekte. Die einfachsten Behaviours sind Animationen und Bewegungen, es existieren aber auch Klassen für Kollisionserkennung, Morphing, Interpolatoren sowie die Möglichkeit, Objekte mit Eingabegeräten zu verbinden. So ließe sich z. B. ein Datenhandschuh mit seiner Repräsentation in dem virtuellen Universum verbinden.

ViewPlatform

Ist an einen BranchGroupNode eine ViewPlatform angehängt, so ermöglicht diese die Abbildung des virtuellen Universums auf den Bildschirm. Da die ViewPlatforms von der Geometrie der angezeigten Objekte unabhängig sind und die Möglichkeit besteht, zwischen mehreren ViewPlatforms zu wechseln, lassen sich die einzelnen Ansichten leicht umschalten.

Appearance

Hier wird die allgemeine Erscheinung Objekt modelliert, also welche Farbe die Oberfläche hat oder ob eine Textur verwendet werden soll.

Light

Die von Java3D unterstützten Lichtmodelle sind: ambient light (allgemeine Beleuchtung), directional light (gerichtetes Licht) und spotlight.

Loader

Um nicht jedes Objekt von Hand mit den von Java3D bereitgestellten Primitiven modellieren zu müssen, stellt Java3D Loader für verschiedene 3D-Modelliertools zur Verfügung, mit deren Hilfe Objekte aus VRML oder 3DSMax einfach eingebunden werden können.

Um mit einer Datenbank zu arbeiten, gibt es für Java die JDBC-Schnittstelle. JDBC heißt Java Database Connectivity und kommt von der Sun-Tochter JavaSoft. JDBC stellt eine Erweiterung der Sprache Java dar und bietet Klassen und Methoden an, die den Zugriff auf beliebige Datenbanken ermöglicht.

Weitere Informationen zu Java3D erhält man auf der Homepage von Sun [14].

4.4.2 Entwicklungssoftware

Zum Entwickeln eigener Java3D-Anwendungen wird die Java Entwicklungsumgebung benötigt. Das JDK (Java Development Kit) kann man bei Sun kostenlos runterladen. Java3D gehört nicht zum Standardpaket des SDK und muss separat runtergeladen werden.

4.4.3 Betrachtersoftware

Für das Ausführen von Javaprogrammen und Java3D-Anwendungen wird eine JRE (Java Runtime Environment) benötigt. Java liegt nicht als ausführbarer Maschinencode vor sondern als plattformunabhängiger Java-Bytecode.

Dieser Java-Bytecode wird zur Laufzeit auf dem jeweiligen Rechner durch einen Java-Interpreter der sogenannten Java Virtual Machine ausgeführt. Dadurch läuft ein Javaprogramm auf allen Systemen. Zwar enthält z.B. der Internet Explorer 6 von Microsoft einen solchen Interpreter, bekannt auch als Microsoft Java Virtual Machine, doch dieser unterstützt kein Java3D. Deshalb muss die JRE von der Firma Sun installiert werden.

4.4.4 Vor- und Nachteile

Java3D ist kein eigenständiges Programm sondern eine Erweiterung für Java. Es besitzt somit alle Vor- und Nachteile der Programmiersprache Java. Java3D ist wie Java plattformunabhängig. Interaktionen und Animationen werden in Java programmiert und können dadurch selbst definiert werden. Man ist nicht auf vordefinierte Interaktionsmöglichkeiten beschränkt.

Mittels der Loader-Funktion können bestehende VRML-Szenen oder 3D-Modelle, die mit einem Modellprogramm erstellt sind in Java3D weitergenutzt werden.

Java ist „Open Source“. Dadurch entstehen keine Kosten für die Entwicklungswerkzeuge.

Für die Erstellung von 3D-Welten mit Java3D sind gute Java Kenntnisse erforderlich.

Eine JRE muss auf dem Rechner installiert werden, damit die Java-Applikation mit dem Java Interpreter ausgeführt werden kann. Ein wesentlicher Nachteil des plattformunabhängigem Java-Bytecodes ist, dass seine Abarbeitung langsam ist.

Durch die komplexe Programmiersprache muss für einen Prototyp lange Entwicklungszeiten eingeplant werden.

Vorteil	Nachteil
Animation und Interaktion frei programmierbar	gute Programmierkenntnisse in Java
plattformunabhängig	Installation einer JRE
Kostenlos	durch die Komplexität von Java sind lange Entwicklungszeiten erforderlich
VRML Szenen können mittels der Loaderfunktion weiterverwendet werden	

Tabelle 4.3: Vor- und Nachteile von Java3D

4.5 VRML

4.5.1 Überblick

VRML steht für Virtual Reality Modelling Language und ist eine Sprache um 3D-Objekte im World Wide Web zu beschreiben. Es ist keine Programmiersprache wie C oder JAVA. Vielmehr ist es ein Dateiformat, das das Aussehen und die Anordnung dreidimensionaler Objekte im Raum beschreibt. Dies erfolgt mittels einer Baumhierarchie, die Szenengraph genannt wird. Die Blätter der Hierarchie (Objektknoten) sind die Gestaltknoten der Szene und damit deren sichtbare Objekte.

Die Äste (Gruppenknoten) und Zweige (Kindknoten) des Baums sind Knoten, die Transformationen wie Verschieben oder Drehen auf die Objekte anwenden oder Abhängigkeiten wie Gruppierungen zwischen Objekten definieren.

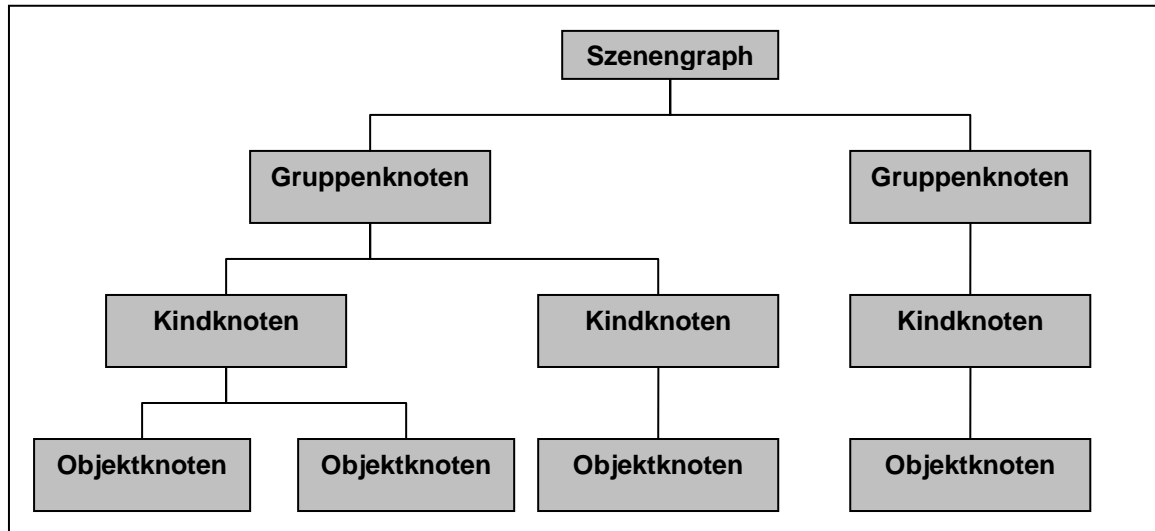


Abbildung 4.6: VRML Szenengraph

So wie beim HTML-Standard sorgt ein Komitee, das Web3D Konsortium [29], früher unter dem Namen VRML Architecture Group (VAG) bekannt, für eine einheitliche Definition und Erweiterung der Sprache.

4.5.1.1 VRML 1.0

Die erste Spezifikation von VRML wurde im Jahr 1995 vorgestellt. VRML 1.0 basiert auf dem Open Inventor 3D-Standard von Silicon Graphics, der Mitte der 90er Jahre eigens zur Darstellung und Manipulation von 3D-Objekten entwickelt wurde. Zwar konnte man mit dieser Version komplexe Szenen mit Materialien, Lichtern und verschiedene Kameraperspektiven erstellen, doch gab es keine Funktionen, die Interaktionen und Animation zuließen. Auch gab es keine Möglichkeit Daten aus oder in eine VRML Szene zu transportieren, da keine externe Programmierschnittstelle integriert wurde.

4.5.1.2 VRML 2.0

Der Wunsch nach Interaktion und Animation floss in die Spezifikation VRML 2.0 ein, die im Jahre 1996 verabschiedet wurde. VRML 2.0 bietet zahlreiche

Möglichkeiten, mit denen einer Szene Dynamik und Interaktion verliehen werden kann. Folgende Erweiterungen wurden in VRML 2.0 eingebaut:

Verbesserte statische Welt

VRML 2.0 unterstützt räumlichen Klang. Durch Einbeziehung dieser Technik kann der Realitätsbezug erhöht werden. Hintergrundgeräusche wie Vogelgezwitscher oder Geräusche, die bei bestimmten Ereignissen erklingen, erzeugen beim Benutzer einen hohen Grad an Immersion.

Boden und Himmel werden getrennt als Hintergrund in der Szene erzeugt. Dadurch lassen sich entfernte Gebirge, Wolken und Nebel nachbilden. Statt flacher Ebenen kann man Geländereliefs als Untergrund verwenden.

Interaktionen

Man kann nun in der Welt direkt mit Objekten oder virtuellen Lebewesen interagieren. So genannte Sensoren erzeugen Ereignisse (EventOut), wenn bestimmte Bereiche der Welt betreten oder angeklickt werden. Diese können wieder andere Prozesse in Gang setzen. Es gibt Time-, Proximity-, Visibility-, Collision- und Touch-Sensoren, welche das Verstreichen einer Zeitspanne, das Annähern des Benutzers, die Sichtbarkeit von Objekten, das Zusammentreffen des Avatars mit einem Objekt bzw. die Berührung mit dem Mauszeiger signalisieren.

Neu in VRML 2.0 ist, dass alle Objekte undurchdringbar sind. Der Benutzer prallt an den festen Objekten ab, wenn er versucht sie zu durchdringen. Durch diese Eigenschaft sind Treppen oder Schrägen in der VRML Welt begehbar.

Animation

Durch Animationen kann in einer Szene die Größe, Position oder das Aussehen eines Objektes verändert werden. Lichter können an- oder ausgeschaltet werden. Geräusche können erklingen oder beendet werden. Für die Animation werden Interpolatoren verwendet. Durch Angabe von Schlüsselwerten bezüglich Position, Orientierung oder etwa Größe eines Objekts berechnen sie zeitgesteuert die notwendigen Zwischenwerte selbst. Dieses Verfahren kennt man auch unter dem Begriff „Keyframe Animation“. Animationen werden meist durch Ereignisse in Gang gesetzt; z.B. erzeugt ein Sensor ein EventOut, wenn die linke Maustaste gedrückt wurde. Dieses Signal wird mittels „Route“ an ein Objekt geleitet. Das Objekt empfängt das Signal (EventIn) und löst damit eine Animation aus. Die Verbindung zwischen einem EventOut und EventIn nennt man „Route“, womit man explizit bestimmt, welches Objekt bei einem Ereignis angesprochen wird.

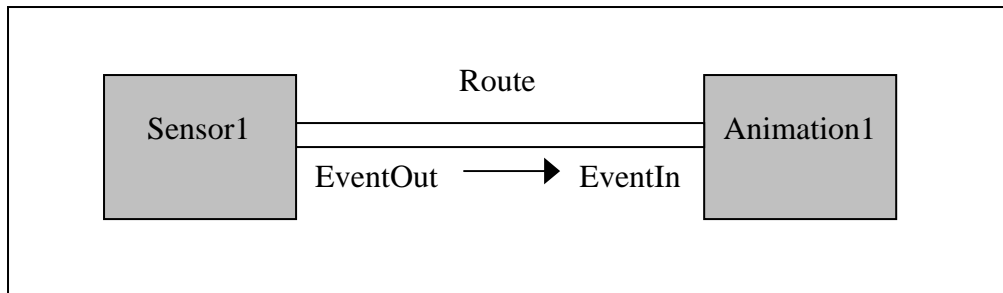


Abbildung 4.7: Route Anweisung

Scripting

Für den Fall, dass die vorgegeben Interaktionsmechanismen in VRML 2.0 nicht ausreichen, gibt es Skriptknoten. Ein Skriptknoten kann Ereignisse empfangen, sie verarbeiten und wiederum neue Ereignisse zur Verarbeitung durch andere Knoten erzeugen. Zudem kann ein Skriptknoten Zustände und Parameter speichern und verwalten. Dadurch ist es möglich zur Laufzeit auf die Szene und ihre Objekte Einfluss zu nehmen. Ferner können VRML-Dateien über Skriptknoten mit anderen, nicht in VRML kodierten Applikationen gekoppelt werden, um anwendungsübergreifenden Datenaustausch zu ermöglichen.

Der Skriptknoten unterstützt die Programmiersprachen Java und Javascript.

Des Weiteren wurde die Syntax grundlegend überarbeitet, wodurch VRML 2.0 nicht mehr abwärtskompatibel ist.

4.5.1.3 VRML 97

Unter dem Begriff VRML 97 wurde VRML im Jahre 1997 zum ISO-Standard (ISO/IEC 14772-1:1997). Zwischen VRML97 und VRML 2.0 gibt es praktisch keinen Unterschied.

4.5.2 Entwicklungssoftware

Der Inhalt einer VRML-Datei besteht aus einem lesbaren Text, der im UTF-8 Format kodiert ist. Das bedeutet, dass eigentlich ein ganz einfacher Texteditor ausreicht um eine VRML-Welt zu erzeugen.

Komfortabler sind dagegen VRML-Builder. Ein Klassiker unter denen ist Cosmo Worlds von Silicon Graphics, einer der Mitbegründer der VRML-Spezifikation. VRML-Builder haben den Vorteil, dass sie eine grafische

Benutzeroberfläche besitzen. Dadurch lassen sich 3D-Welten mit Drag und Drop erstellen, wobei man sich nicht um die Syntax kümmern muss. VRML-BUILDER besitzen den Nachteil, dass sich komplexe 3D-Objekte nur schwierig erstellen lassen. Für ihre Erstellung werden professionelle 3D-Modellierwerkzeuge benötigt. Eines der bekanntesten Programme ist 3D Studio Max von der Firma Discreet. Dieses Programm besitzt spezielle Funktionen und Eingabemasken um die Erstellung von komplexen Objekten zu vereinfachen. Durch eine Exportfunktion lassen sich die Objekte im VRML-Dateiformat abspeichern und in eine vorhandene Szene integrieren. In Abbildung 4.8 sehen wir die Benutzeroberfläche von Cosmo Worlds.

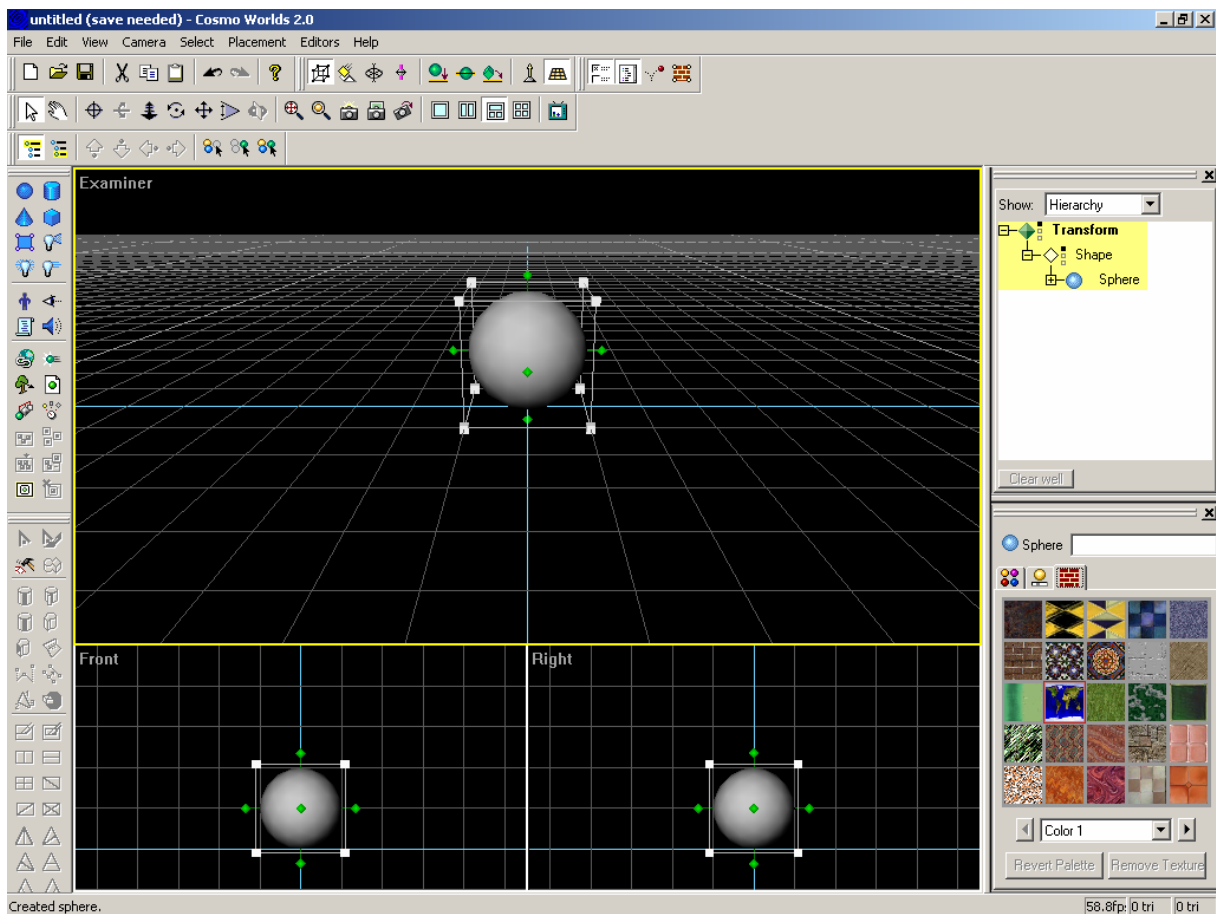


Abbildung 4.8: VRML-BUILDER Cosmo World

Der dazugehörige VRML Code sieht wie folgt aus:

```
#VRML V2.0 utf8
Transform {
  Children [
    geometry      Shape {
      Sphere { #Erzeugt eine Kugel
    }
  ]
  translation    0 1 0 #Verschiebung der Kugel
}
```

Abbildung 4.9: VRML Code

Auf der linken Seite der Abbildung befindet sich die Werkzeugleiste. Primitiven wie Kugel, Zylinder, Kegel und Quader lassen sich erzeugen und nach dem Bauklötzchenprinzip zu einfachen 3D Figuren zusammensetzen. In der Mitte sieht man die 3D Welt. Ansichten wie z.B. Seiten- oder Draufsicht können mit der oberen Menüleiste eingestellt werden. Auf der rechten Seite befindet sich das Texturwerkzeug. Per Mausklick wird die gewählte Textur dem 3D Objekt zugeordnet. Ein zusätzliches Fenster zeigt die aktuelle VRML-Szene in Form eines Baums mit ihren Knoten und Blättern.

4.5.3 Betrachtersoftware

Damit eine VRML-Welt betrachtet werden kann, muss ein Plugin installiert werden. Es gibt eine Vielzahl von VRML-Plugins für verschiedene Betriebssysteme, u.a. den Contact von der Firma Blaxxun, Cortona von Parallegraphics und Cosmo Player von Silicon Graphics. Bei der Wahl des Plugins sollte man die aktuelleren nehmen, da sie die Funktionen von 3D-Grafikkarten besser unterstützen.

4.5.4 Vor- und Nachteile

Mit VRML-Builder lassen sich mit wenigen Schritten komplette 3D-Welten erstellen auch ohne Kenntnisse der VRML Syntax. Sie bieten ausreichend Funktionen, um Animationen und Interaktionen zu erstellen.

Über das Scripting lassen sich zusätzliche Animationen und Interaktionen programmieren. Außerdem kann über das Scripting ein Datenaustausch mit nicht VRML Applikationen realisiert werden. Durch die einfache Syntax von VRML lassen sich in kurzer Zeit Prototypen erstellen.

Um VRML-Welten betrachten zu können wird ein Plugin für den Internetbrowser benötigt. Komplexe Szenen lassen sich mit den Buildern nur kaum konstruieren, sie müssen mit 3D-Modellierprogrammen erstellt werden. Solche professionellen Modellierprogramme kosten mehrere tausend Euros.

Vorteil	Nachteil
3D Szenen lassen mit VRML-Builder schnell erstellen	benötigt ein Plugin
durch Scripting kann man Einfluss auf Szene nehmen und Datenaustausch mit andern Applikationen vornehmen	komplizierte Formen müssen mit einem 3D Modellierprogramm erstellt werden
durch die einfache Syntax, können in kürzester Zeit Prototypen entstehen	

Tabelle 4.4: Vor- und Nachteile von VRML

4.6 Extensible 3D

4.6.1 Überblick

Das Extensible 3D (kurz X3D) Format wird von der X3D Working Group des Web3D [29] Konsortium entworfen, implementiert und evaluiert. Die Ziele bei der Entwicklung von X3D sind die folgenden:

- Kompatibilität zu VRML97, dabei speziell in Hinsicht auf Applikationen, Player und Werkzeuge
- Der X3D-Kern (Core X3D) soll kompakter sein als der Kern von VRML97. Er soll Funktionen zur Realisierung von eCommerce- und Unterhaltungsanwendungen sowie zum verteilten Lernen bieten
- Das schlanke Core-Profil soll einen Erweiterungsmechanismus bieten, der es ermöglicht, neue Komponenten zu integrieren
- Die Unterstützung von anderen Kodierungen wie z.B. XML

Es gibt momentan sechs Profiles, die in der X3D Document Type Definition (kurz DTD) aufgeführt werden. Jedes dieser Profile ist für ein spezielles Anwendungsgebiet konzipiert.

1. die eigentliche Umsetzung von VRML97
2. geografische Visualisierung mit GeoVRML

-
3. H-Anim dient zur Darstellung von Menschen
 4. Verwendung von NURBS
 5. Disjava VRML zur Nutzung von verteilten interaktiven Simulationen
 6. LatticeXvl zur kompakten Übertragung komplexer Geometrien

Das X3D-Format zeichnet sich durch folgende Merkmale aus:

Der eigentliche X3D-Kern, der nur die notwendigsten Spezifikation enthält, kann durch Anlegen weitere Profile erweitert werden. Dadurch kann auf die speziellen Bedürfnisse der einzelnen Anwender eingegangen werden. So wird ein Biologe ganz andere Anforderungen an eine dreidimensionale Darstellung haben als ein Geograph. X3D lässt sich dadurch fast unbegrenzt erweitern.

Durch das Scene Authoring Interface (SAI) soll die Kommunikation zwischen einem externen Applet innerhalb einer HTML Seite und der Szene ermöglicht werden. Es ist ein API, das den Zugriff auf den Szenengraph ermöglicht und außerdem die Möglichkeit bietet, Verhaltensweise für die Szenen zu definieren. Die SAI unterstützt momentan Java und Javascript. Über diese SAI kann wie bei VRML eine Datenbank genutzt werden.

Die Kodierung der Zeichen erfolgt im XML-Format. Die Struktur wird durch eine XML-Grammatik definiert.

Ein großes Manko von VRML war das Fehlen eines binären Dateiformats. Das soll sich zukünftig beim X3D-Format ändern. In Planung ist ein binäres Dateiformat, welches zu der Struktur des UTF-8 Formates kompatibel ist und weniger Speicherplatz zum Ablegen der Daten benötigt sowie Streaming fähig ist.

4.6.2 Entwicklungssoftware

Für die Erstellung einer 3D-Welt mit X3D stellt das Web3D Konsortium das Tool „X3D Edit“ zur Verfügung. Damit der Editor funktioniert, müssen sowohl Java als auch der XML-Parser Xena von IBM installiert sein. Das Web3D-Konsortium bietet beide Komponenten inklusive einer Installationsanweisung auf seinen Webseiten an.

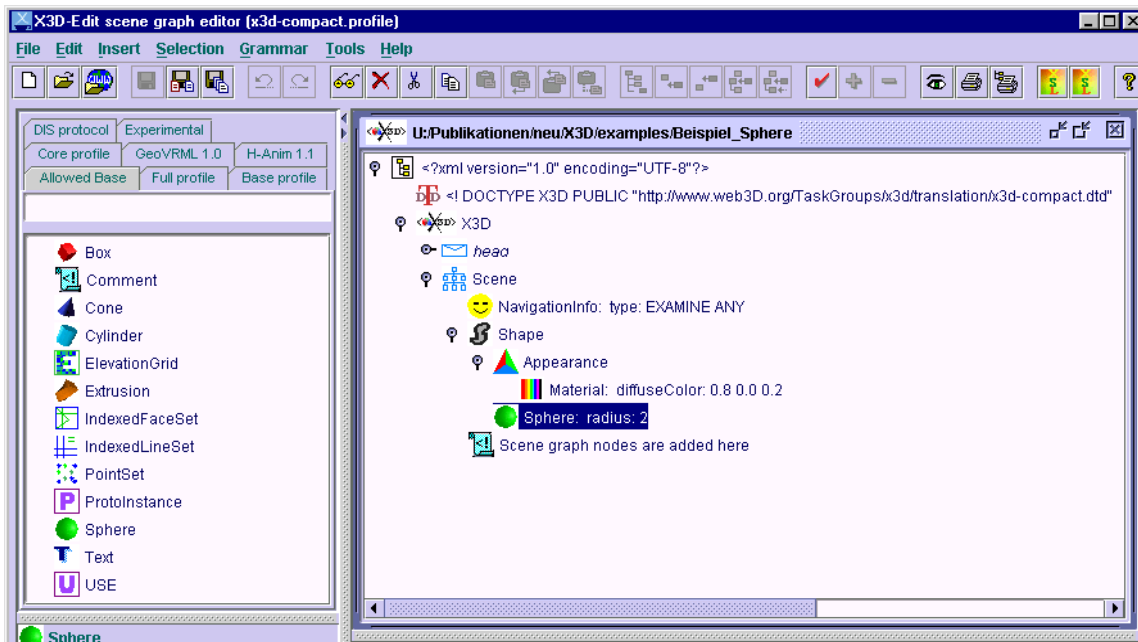


Abbildung 4.10: Benutzeroberfläche von X3D Edit

Die Abbildung 4.10 zeigt die graphische Darstellung des Quellcodes. In dieser Ansicht unterscheidet sich X3D äußerlich kaum von VRML97. Neu sind einige typische XML-Eigenschaften wie etwa die eingebundene DTD (Document Type Definition). Der Quellcode in Textform sieht wie folgt aus:

```

<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE X3D PUBLIC "http://www.web3d.org/TaskGroups/x3d/ translation/x3d-compact.dtd"
  <X3D>
    <head>
      <meta content="X3D-Edit, http://www.web3d.org/TaskGroups/x3d/ translation/README.X3D-Edit.html"
        name="generator"/>
    </head>
    <Scene>
      <Shape>
        <Appearance>
          <Material diffuseColor="0.8 0.0 0.2"/>
        </Appearance>
        <Sphere radius="2"/>
      </Shape>
    </Scene>
  </X3D>

```

Abbildung 4.11: X3D Quellcode

Die Datei beginnt mit einer DTD, die die Version und Sprache festlegt. Anschließend folgt der Head, der wie bei HTML in der Regel Meta-Angaben beinhaltet. Die Beschreibung der Szene beginnt beim <Scene> Tag. Man sieht, dass zwar die gleichen Bezeichner verwendet werden, aber die Syntax anders ist. Da X3D in XML codiert wird, fällt die Konvertierung mittels eines XSL-Stylesheets in andere Formate leicht. Das Web3D-Konsortium legt das entsprechende Stylesheet "X3dToVrml97.xsl" und „Vrml97ToX3d“ dem X3D-

Edit bei. Damit kann ein vorhandener Vrm197 Quelltext in X3D umgewandelt werden bzw. von X3d wieder in Vrm197 zurückgewandelt werden.

4.6.3 Betrachtersoftware

Für das Betrachten wird ein X3D-fähiger Viewer benötigt. Auf der Seite von Web3D[15] wird der Xj3D angeboten. Xj3D ist ein open source Projekt von Sun und wird kostenlos zum Herunterladen angeboten. Firmen wie Blaxxun [30] oder Parallelgraphics [31] bieten auf Ihre Homepage bereits X3D Viewer an.

4.6.4 Vor- und Nachteile

Durch die Anwendung von XML ergeben sich folgende Vorteile.

Inhalt und Aussehen werden strikt getrennt. Mit XML lassen sich die verschiedensten Elemente beschreiben. Es bedarf lediglich einer entsprechenden Formulierung der DTD. Durch eine entsprechende Erweiterung der DTD kann man den Sprachumfang jederzeit ausbauen. Da X3D eine Teilmenge von VRML97 enthält, ist Abwärtskompatibilität gewährt. Durch Einsatz des XSL-Stylesheets können VRML97-Quelltexte in das neue X3D-Format umgewandelt werden.

X3D soll bald zum weltweitem ISO Standard werden. Doch bis X3D zum endgültigen internationalen Standard berufen wird, kann die jetzige Spezifikation noch Veränderungen unterliegen.

Vorteil	Nachteil
Kompatibilität zu VRML97-Anwendungen	endgültige Spezifikation steht noch nicht fest
VRML97-Code lässt sich leicht in X3D-Code umwandeln	
Erweiterbarkeit durch Profile	
Software ist kostenlos	
Benötigt weniger Speicherplatz	
Streaming fähig	

Tabelle 4.5: Vor- und Nachteile von X3D

5 Entwicklung des Systems

5.1 Analysephase

Eine Analyse wird benötigt um Auskunft über die Funktionalitäten des Systems zu erhalten. Angefangen von einer Ist-Analyse, die die Ausgangslage beschreibt werden bei der Soll-Analyse die Ziele und die darausfolgenden Anforderungen des Systems beschrieben. Anschließend wird im Rahmen der Machbarkeitsanalyse das Konzept vorgestellt, mit dem das System realisiert werden soll.

5.1.1 Ausgangssituation

Das Zwangsarbeiterlager in Berlin Niederschöneweide liegt heute im Bezirk Treptow Köpenick in der Britzer Straße. Der Krieg hat zwar starke Spuren auf dem Gelände hinterlassen, es ist mit seinen Anlagen im Großen und Ganzen aber noch gut erhalten. Das Betreten des Geländes ist nur im Rahmen einer Führung möglich, die die Berliner Geschichtswerkstatt für Gruppen und zu bestimmten Anlässen nach Absprache anbietet.

Die Geschichtswerkstatt ist in Besitz von authentischen Dokumenten, die das Gelände mit seinen Anlagen vor 60 Jahren beschreibt. Diese Dokumente enthalten Baupläne, Skizzen, Fotos und Briefe. Die Herkunft dieser Dokumente ist unterschiedlich. Als Quellen sind das Bauaktenarchiv Treptow, die Senatsverwaltung für Stadtentwicklung, das Bundesarchiv und das Fotoarchiv der Berliner Geschichtswerkstatt zu nennen. Mit Hilfe der Unterlagen kann man sich leider nur schwer räumlich vorstellen wie das Zwangsarbeiterlager vor 60 Jahren ausgesehen hat.

Deshalb hat eine Gruppe von Studenten der Fachhochschule für Technik und Wirtschaft in Berlin Karlshorst anhand dieser Dokumente eine 3D-Rekonstruktion des Lagers erstellt. Als Ergebnis dieser Projektarbeit entstanden drei Szenen. Eine Szene zeigt das gesamte Lagergelände, die zweite eine Innenansicht einer Baracke und die dritte eine Ansicht eines Luftschutzkellers. Die Szenen wurden mit Hilfe des 3D-Modellierprogramms 3D Studio MAX erstellt. Die Objekte wurden erst einzeln modelliert und später nach dem Baukastenprinzip zu einer Gesamtszene zusammengesetzt. Anschließend wurde eine 3D-Visualisierungssoftware eingesetzt um die Szenen begebar zu machen.

5.1.2 Zielsetzung

Das Ziel dieser Arbeit ist die Integration von historischen Informationsdaten in Form von Texten und Bildern in eine 3D-Welt. Die Benutzer können mit dem zu entwickelnden System das Gelände als 3D-Darstellung erkunden. Informationen zu den Objekten werden eingeblendet und beschreiben, welche historischen Quellen zur Rekonstruktion der 3D Objekts beigetragen haben. Mit dem System können Historiker sich ein Bild davon machen, wie das Gelände vor ungefähr 60 Jahren ausgesehen hat. Die Navigation in den virtuellen Welten sollte intuitiv sein, damit die Hemmschwelle im Umgang mit dem Medium Computer abgebaut wird. Zielgruppe des Systems sind vor allem Geschichtsinteressierte, die sich mit diesem Thema beschäftigen. Aber auch Neugierige können mit dem System angesprochen werden und in Ihnen eventuell das Interesse für diese Thematik zu wecken. Mit dem System können Besucher sich vorab informieren, wie das Gelände vor 60 Jahren ausgesehen hat, bevor Sie das heutige Gelände bei einer Besichtigungstour erkunden.

5.1.3 Anforderungen

Der Benutzer hat die Wahl zwischen drei Szenen. In jeder Szene kann der Benutzer sich frei bewegen und die Objekte aus verschiedenen Ansichten betrachten.

Informationen, die die Objekte beschreiben, sollen über separate Fenster eingeblendet werden.

Außerdem sollte eine Dokumentenebene eingerichtet werden, wo die Möglichkeit besteht alle verfügbaren Dokumente und Annotationen separat anschauen zu können. Falls der Wunsch besteht, das betreffende Objekt in der 3D Welt anschauen zu wollen, führt ein Link zur der betreffenden Szene.

Das System sollte autorisierten Personen die Möglichkeit bieten, über eine grafische Benutzeroberfläche, Szenen anzulegen, zu ändern der zu löschen. Einfügen, ändern oder löschen von Annotationen und Dokumenten sollte ebenfalls über die grafische Oberfläche möglich sein. Alle notwendigen Daten, die für die Präsentation der Szenen notwendig sind, sollen über eine Datenbank erfasst werden.

Da historische Dokumente manchmal ungenaue Angaben beinhalten, sollte noch die Möglichkeit bestehen, die Position der Objekte über eine grafische Benutzeroberfläche nachträglich zu ändern.

Des Weiteren sollte die 3D-Rekonstruktion des Lagers im Internet präsentiert werden, damit das System möglichst eine breite Masse von Internetbenutzern erreicht.

5.1.4 Machbarkeit

In Kapitel 3 wurden die gängigsten 3D-Visualisierungstechniken vorgestellt. Für das zu entwickelnde System kommt die VRML97-Technologie zum Einsatz. Die Technologie bietet für unseren Zweck ausreichende Interaktions- und Animationsmöglichkeiten. Über den Skriptknoten können Daten zwischen einer VRML-Szene und einer HTML-Seite ausgetauscht werden. Die 3D-Modelle, die mit dem Modellierprogramm erstellt worden sind, lassen sich mit der integrierten Exportfunktion des Programms in das VRML97-Datenformat umwandeln. Außerdem gibt es eine Vielzahl von VRML-Plugins, die für fast jedes Betriebssystem und fast jeden Internetbrowser erhältlich sind.

Die Shockwave-3D-Technologie wird nicht genutzt, da die 3D-Szenen zuviel Speicherplatz benötigen. Die begehbaren 3D-Szenen, die die Projektgruppe mit Director erstellt haben, haben eine Größe von etwa 30 MBytes. Würde man die Szenen für die Internetpräsentation in Shockwave3D umwandeln, sind lange Ladezeiten im Internet vorprogrammiert.

3D-Engines versprechen, realistische Welten darzustellen, aber die Benutzung solcher Engines verlangt gute Programmierkenntnisse sowie Kenntnisse über die Funktionsweise der 3D-Engine. Schlechte Dokumentationen erschweren außerdem die Benutzung der 3D-Engines. Die Datenbankbindung erfolgt nur durch den Einsatz einer zusätzlichen Programmierschnittstelle. Die Darstellung der 3D-Welt ist mit hohem Rechenaufwand verbunden. Deshalb wird die 3D-Engine nicht verwendet.

Durch die Komplexität der Programmiersprache Java können in kurzer Zeit keine Prototypen erstellt werden. Die Datenbankbindung erfolgt wie bei der 3D-Engine nur durch den Einsatz einer zusätzlichen Programmierschnittstelle. Die langsame Abarbeitung des Java Bytecodes hat Auswirkung auf die Performance der 3D-Anwendung. Dadurch können bei großen 3D-Anwendungen zu langen Wartezeiten führen. Aus diesen Gründen wird Java3D nicht verwendet.

Die X3D-Technologie, die bald VRML97 ablösen soll, befindet sich noch in der Entwicklungsphase und kommt deshalb nicht zum Einsatz.

Damit das System über das Internet abrufbar ist, wird ein Webserver benötigt. Die Hauptaufgabe eines Webserver besteht in der Verteilung von angeforderten Dokumenten. Greift ein Benutzer auf die Site zu, sucht der Webserver die gewünschte Datei und sendet sie dem Rechner, der sie angefordert hat.

Der Webserver Apache ist einer der bekanntesten Webserver. Einer Studie von Netcraft [32] hat gezeigt, dass Apache immer öfter als Webserver eingesetzt wird. Die Vorteile von Apache liegen auf der Hand. Apache ist ein Open Source Projekt und somit kostenlos. Des Weiteren kümmert sich die Projektgruppe um die weitere Entwicklung der Software. Sicherheitslücken werden schnellstmöglich durch Softwareupdates behoben. Apache läuft sowohl auf Linux als auch unter Windows. Neben dem Apache gibt es noch den Internet Information Services von der Firma Microsoft. IIS läuft nur auf Windows basierten Betriebssystemen und wird ab dem Betriebssystem Windows 2000 beigelegt.

Für diese Arbeit wird der Apache Webserver eingesetzt, da die Software kostenlos ist und auf den gängigsten Betriebssystemen läuft.

Als Datenbanksystem wird MySQL [33] verwendet um die Dokumente und 3D Modelle zu verwalten. MySQL ist ein Open Source Projekt und wird kostenlos zum Download angeboten. Die Datenbank ist für Linux- und Windows-Systeme verfügbar und arbeitet mit dem Webserver Apache gut zusammen.

VRML selbst unterstützt keine Datenbanken. Aber mit Hilfe des Skriptknotens hat man die Möglichkeit über die Programmiersprache Java und JDBC (siehe Kapitel 4.4) Zugriff auf eine Datenbank zu erhalten. So wäre es denkbar, dass z.B. ein Ereignis in der VRML Szene ein Java Applet dazu veranlasst Informationen aus einer Datenbank abzurufen und dann diese Informationen zur Laufzeit in der 3D Welt angezeigt werden. Objekte in der VRML-Welt können mit einem Java-Applet nach Bedarf ein- oder ausgeblendet werden ohne die Szene neu laden zu müssen.

Eine andere Möglichkeit wäre statt der Verwendung von Java und JDBC eine Skriptsprache wie PHP oder ASP. PHP ist die Abkürzung für Personal Home Page Tool und ist wie Active Server Pages (ASP) eine serverseitige Skriptsprache. Mit diesen Sprachen lassen sich in kürzester Zeit dynamische Webseiten erstellen. Meist wird die Skriptsprache in Zusammenhang mit einer Datenbank genutzt. Ruft ein Besucher eine Seite auf dem Server auf, so sucht sich das Skript die angeforderten Informationen aus der Datenbank heraus und erstellt die Seite zusammen. Anschließend wird diese Seite an den Empfänger geschickt. Die Skriptsprache ASP funktioniert nur in Verbindung mit dem IIS. PHP dagegen läuft sowohl auf einem Apache als auch auf einem IIS.

Um aus einer VRML Welt mittels einer Skriptsprache auf eine Datenbank zuzugreifen, wäre folgender Lösungsweg denkbar. Ein Ereignis in der VRML-Welt aktiviert über den Skriptknoten ein PHP-Skript auf dem Server. Dieses greift mit den implementierten Datenbankfunktionen auf eine Datenbank zu. Das Resultat wird zum Rechner geschickt und dann z.B. außerhalb der VRML-Szene in einem HTML-Fenster angezeigt.

Die Lösung mit der Skriptsprache ist für diese Arbeit ausreichend. Die Gründe sind, dass mit PHP in kurzer Zeit Prototypen erstellt werden können und die Datenbankfunktionen schon implementiert sind. Außerdem muss keine zusätzliche Software für den Browser installiert werden, da das Skript auf der Serverseite ausgeführt wird. Ein Java-Applet in Verbindung mit JDBC hat den Nachteil, dass die Applikation nicht sehr schnell läuft und der Browser außerdem einen gewissen Grad Java unterstützen muss.

5.2 Export von 3D Studio Max nach VRML

Dieser Abschnitt beschreibt den Vorgang, wie man die vorhandenen 3D-Modelle in ein VRML97-Dateiformat umwandelt.

Wie schon in Kapitel 5.1.1 beschrieben, wurde das 3D Modellierprogramm 3D Studio MAX zum Erstellen der 3D-Objekte benutzt. Das Programm enthält eine Exportfunktion, die es erlaubt, Objekte in verschiedene Dateiformate zu exportieren. Um die Exportfunktion zu demonstrieren, wird ein Tisch in das VRML97-Dateiformat umgewandelt. Als erstes wird das Modell geladen.

Die Abbildung 5.1 zeigt die Benutzeroberfläche des Programms. Man hat insgesamt vier Fenster. Jedes dieser Fenster zeigt das Objekt aus verschiedenen Ansichten. Der Tisch besteht aus insgesamt sechs Einzelobjekten. Eine Tischplatte, das Seitenteil und vier Beine. Jedes Einzelobjekt wurde mit einem Namen versehen.

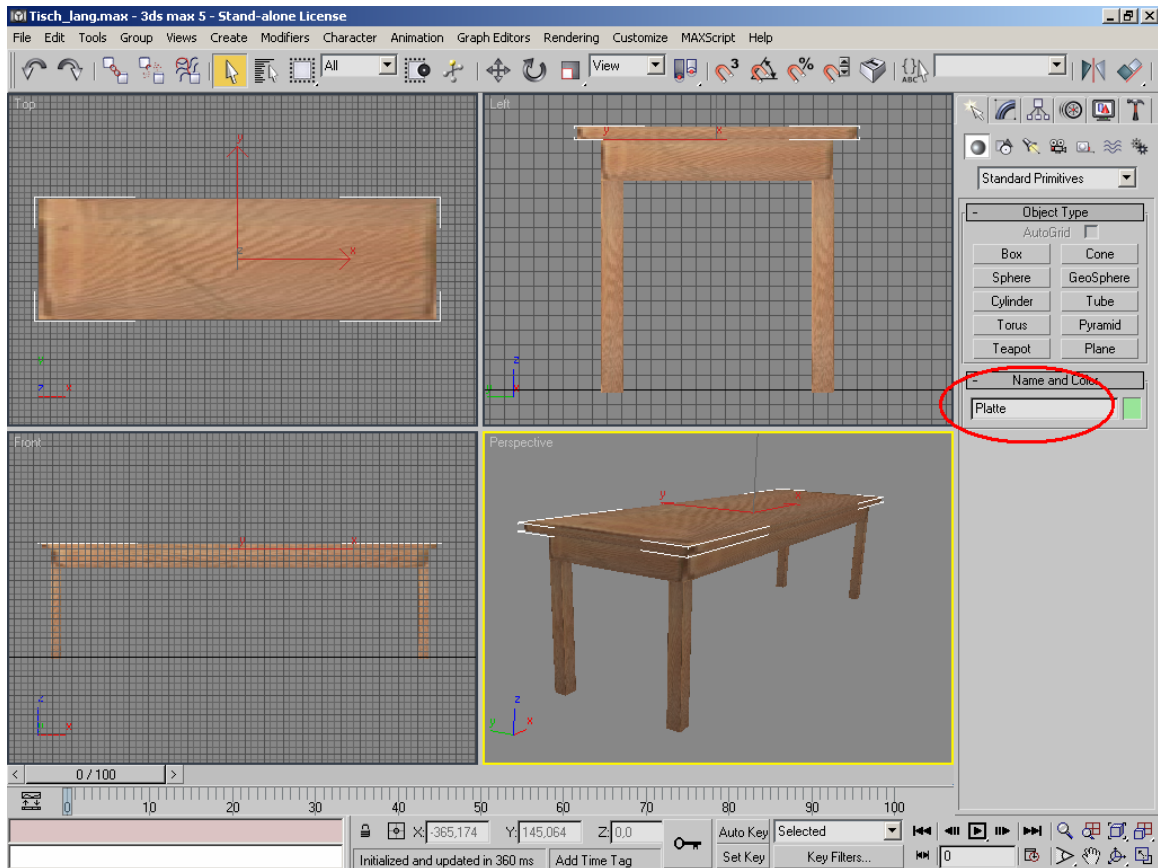


Abbildung 5.1: Benutzeroberfläche von 3D Studio Max

Anschließend ruft man die Exportfunktion über den Menüpunkt File auf. Hier gibt man neben dem Dateinamen noch das gewünschte Dateiformat an.

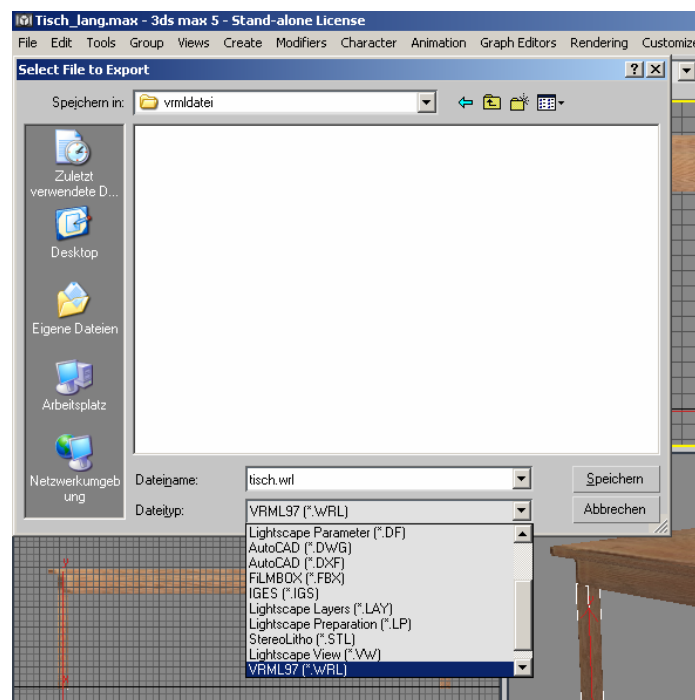


Abbildung 5.2: Export nach VRML

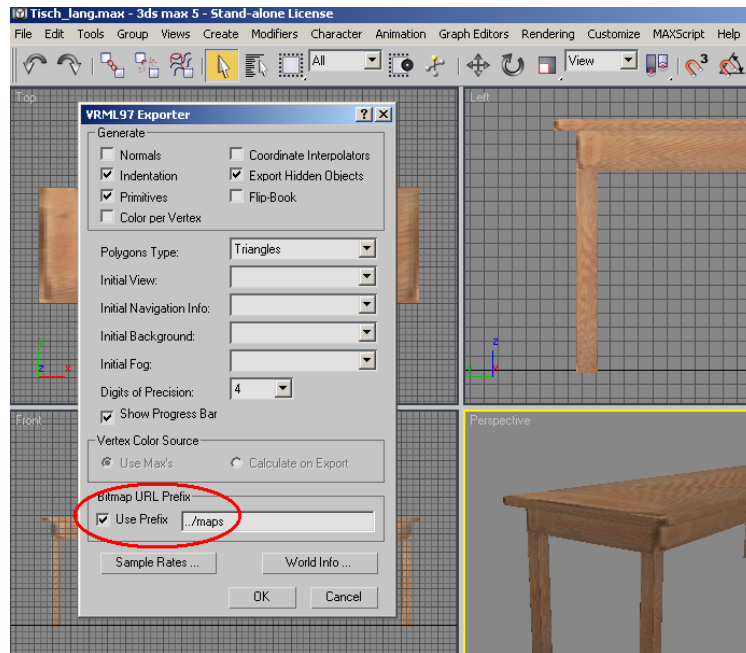


Abbildung 5.3: Exportfunktion von 3D Studio Max

In der Abbildung 5.3 sieht man das Exportfenster. Hier können einige Einstellungen vorgenommen werden wie z.B. in welchem Verzeichnis die Texturen liegen. In unserem Fall befinden sich die Texturen im gleichen Verzeichnis, wir ändern entsprechend den Eintrag. Ansonsten lassen wir die Einstellungen und bestätigen den Export mit OK. Das Resultat des Exportiervorganges kann man mit einem Browser und dem notwendigen VRML-Plugin betrachten.

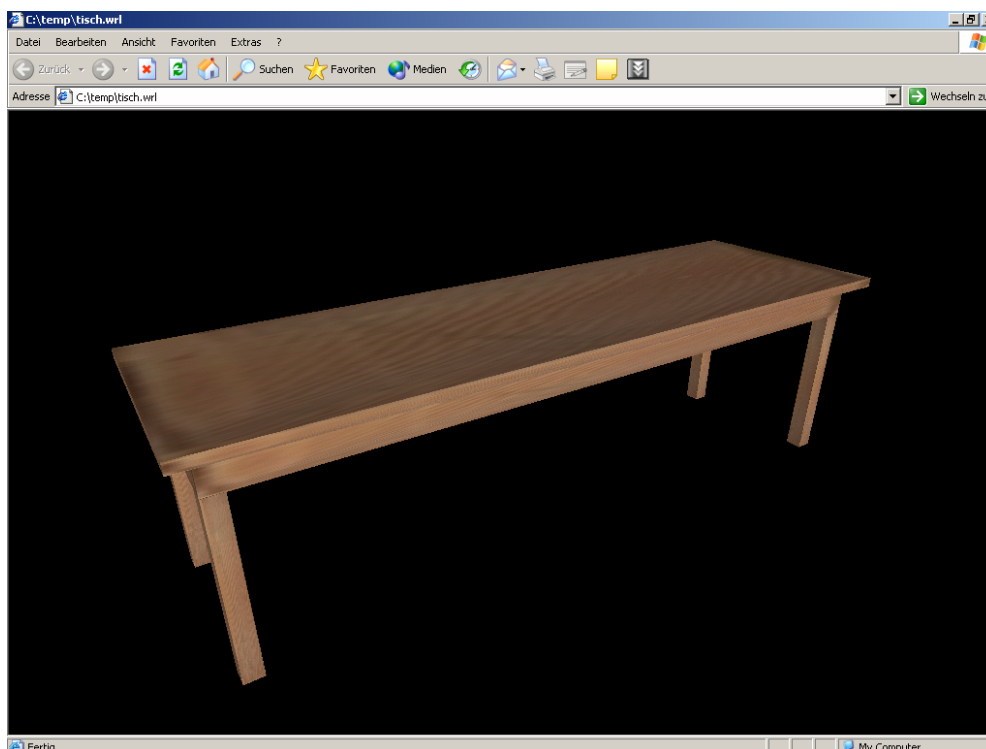


Abbildung 5.4: Ergebnis des Exports mit VRML plugin betrachtet

Dieser Vorgang muss für jedes 3D Objekt durchgeführt werden, um später eine Optimierung (Kapitel 5.3) durchführen zu können.

Außerdem muss die Gesamtszenen exportiert werden, um später daraus die Positionen aller Objekte zu erhalten. Die Positionswerte werden später in eine Datenbank eingetragen. Beim Exportieren der Gesamtszene ist darauf zu achten, dass jedes aufgeführte 3D-Objekt der Szene einen eindeutigen Namen hat, damit die Objekte später wieder gefunden werden können.

5.3 Optimierung der VRML Datei

Um die Übersichtlichkeit im VRML-Quelltext zu erhöhen, kann man den einzelnen Objekten Namen vergeben. Dies geschieht mit Hilfe der Anweisung DEF, mit der sich jedem beliebigen Knoten eine Bezeichnung zuordnen lässt.

Die DEF Anweisung haben wir schon im vorherigen Abschnitt benutzt. Wenn wir im Modellierprogramm Namen für die einzelnen Teilobjekte des Tisches vergeben, werden beim Export nach VRML97 die einzelnen Namen mit der DEF Anweisung dem jeweiligen Objekt zugeordnet. Neben der Übersichtlichkeit kann die Anweisung DEF Referenzierungen ermöglichen. Dies wird am folgenden Beispiel verdeutlicht.

```
#VRML V2.0 utf8

Transform {
  children [
    DEF Ball Shape { #dieser Objektknoten heisst Ball
      geometry Sphere { radius 2 } #definiert eine Kugel mit Radius 2
    }
  ]
  translation 0 1 0 #verschiebt den Ball um diese Koordinaten
}
```

Abbildung 5.5: VRML Code mit DEF Anweisung

Dieser Quellcode beschreibt eine Kugel mit dem Radius 2 und den Raumkoordinaten $x=0$, $y=1$ und $z=0$.

Mit der DEF Anweisung wird der Kugel der Name „Ball“ zugewiesen.

Soll die identische Kugel an einer anderen Position nochmals verwendet werden, kann man die Anweisung USE nutzen.

```

#VRML V2.0 utf8

Transform {
  children [
    DEF Ball Shape { #dieser Objektknoten heisst Ball
      geometry Sphere { radius 2 } #definiert eine Kugel mit Radius 2
    }
  ]
  translation 0 1 0 #verschiebt den Ball um diese Koordinaten
}

Transform {
  children [
    USE Ball #Referenz auf Ball Objekt
  ]
  translation 1 0 0
}

```

Die zweite Kugel ist lediglich eine Referenz von Ball und keine Kopie. Würde man den Radius des Objekts Ball ändern, so würde es Auswirkungen auf die zweite Kugel haben.

Abbildung 5.6: VRML Code mit DEF und USE Anweisung

Ein Objekt wird also nur einmal Komplette beschrieben und mit einem Namen versehen. Anschließend können beliebig viele Referenzen auf das Objekt hergestellt werden. Mit dieser Technik kann der Umfang einer VRML-Datei deutlich reduziert werden. Besonders bei allen Barackenmodellen, wo das Fenster oftmals wieder verwendet wird, hat sich diese Technik bewährt. So konnte eine VRML-Datei mit einer Größe von 285 KByte auf die Größe von 52Kbyte reduziert werden.

5.4 Datenbank

5.4.1 Datenbankentwurf

Eine Datenbank ist die elektronische Form eines Karteikastens in der eine Sammlung von Daten, die aus der Sicht des Benutzers zusammengehören, gespeichert wird. Die Datenbank wird üblicherweise von einem Datenbankverwaltungssystem (DBMS) verwaltet. Ein DBMS zusammen mit einer oder mehreren Datenbanken nennt man auch Datenbanksystem. Das grundlegende Element einer Datenbank ist der Datensatz (er entspricht einer Karteikarte). Aus einer gewissen Anzahl von Datensätzen wird eine Tabelle gebildet. Mehrere Tabellen, die zu einer Gesamtheit zusammengefasst und untereinander verknüpft werden, bilden eine Datenbank.

Das zu entwickelnde System benutzt eine MySQL Datenbank, um Dokumente und 3D Szenen zu verwalten. Es können neue Daten aufgenommen, verändert oder entfernt werden. Folgende Daten werden in der Datenbank festgehalten:

- die Szenen mit den 3D Modellen
- die Positionen der 3D Modelle in der Szene
- die Viewpoints der Szene mit deren Annotation
- die Lichtquellen der Szene
- die 3D Modelle mit deren Texturen
- die historischen Dokumente mit Quellenangabe
- das Administrator Passwort

5.4.2 Entity Relationship Modell

Zur Erstellung der Datenbankstruktur wird die Software Powerdesigner von der Firma Sybase [34] verwendet. Als erstes wird ein Entity-Relationship-Modell, kurz ER-Modell erstellt. Das ER-Modell dient dazu, im Rahmen der Datenmodellierung, die reale Welt semantisch präzise zu beschreiben, um sie in Tabellen und Beziehungen innerhalb einer Datenbank abbilden zu können.

Die Abbildung 5.7 zeigt das ER-Modell. Jede Entity besitzt Felder mit dem dazugehörigen Datentyp. Die Entity mit dem Namen 3dmodelle besitzt die Attribute Indexdreid vom Datentyp Integer, Name vom Datentyp Character, Beschreibung vom Datentyp Text und Dateiname vom Datentyp Character.

Durch den Identifier, auch Primärschlüssel genannt, wird jeder einzelner Datensatz eindeutig identifiziert.

Zwischen den Entities gibt es Beziehungen, die in Form von „n:m“, „1:n“ und „1:1“ dargestellt werden. So herrscht zwischen den Entities 3dmodelle und positionundrotation eine 1:n Beziehung. Das bedeutet, dass jeder Positionsangabe ein 3D-Objekt zugeordnet ist. Ein 3D-Objekt kann jedoch auch mehrere Positionsangaben besitzen.

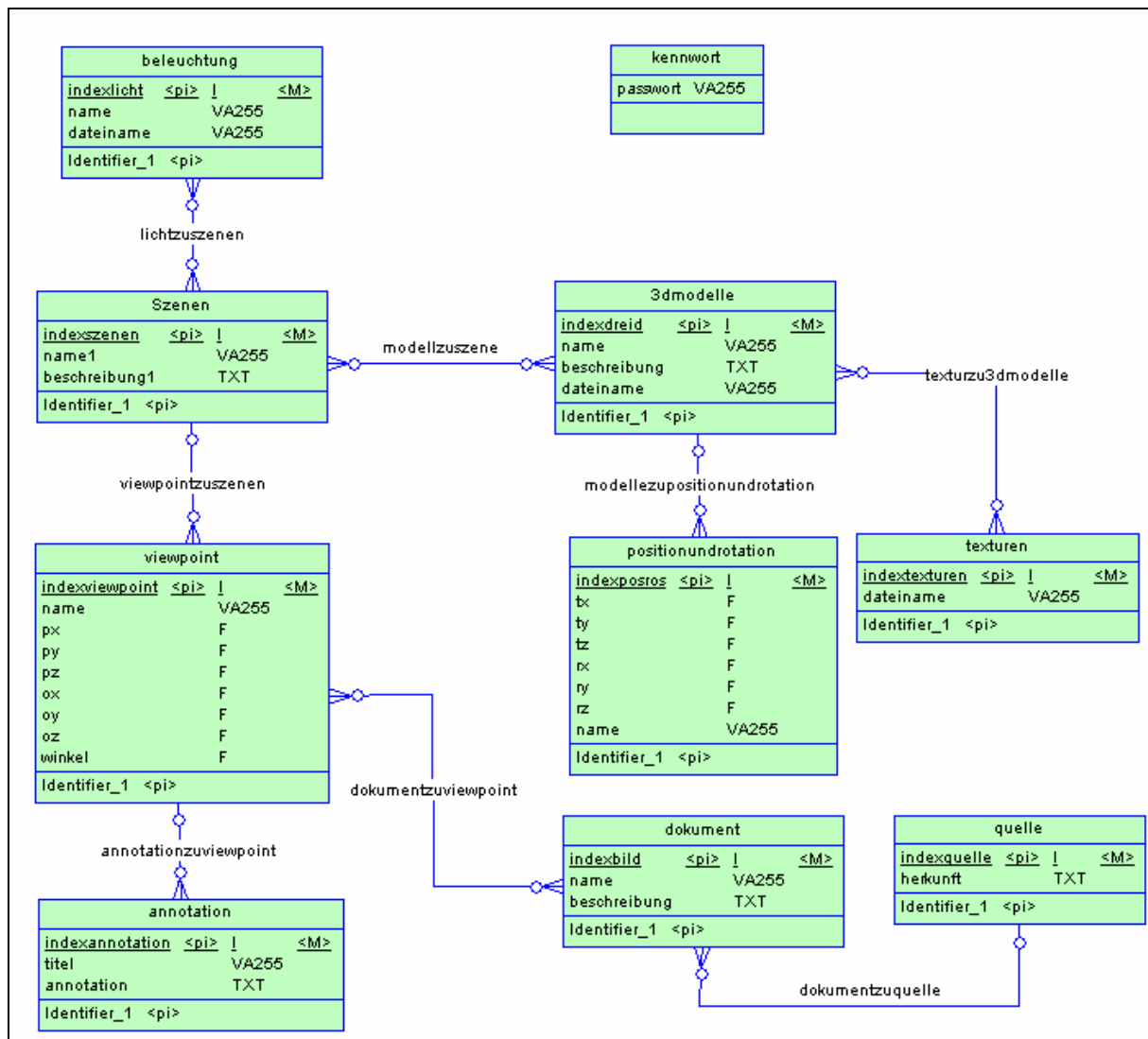
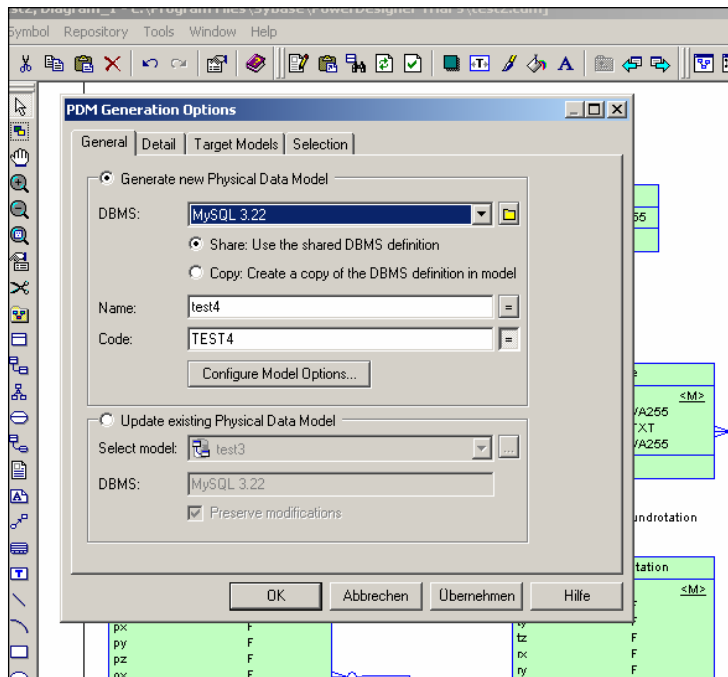


Abbildung 5.7: Entity Relation Ship Modell

5.4.3 Physisches Modell

Nachdem das Modell fertig ist, kann mit dem Powerdesigner das ER-Modell in ein physisches Modell umgewandelt werden. Das physische Modell ist das Endergebnis der Datenbankmodellierung und gibt die tatsächliche Struktur der Datenbank wieder.

Beim Umwandeln in das physische Modell wird das ER-Modell dem Normalisierungsprozess unterzogen. Dabei wird das Gesamtmodell mit Hilfe eines Regelwerks so bearbeitet, dass Informationen redundanzfrei abgespeichert werden.



In der Abbildung 5.9 sieht man die Eingabemaske für die Umwandlung des ER-Modells.

Unter DBMS kann man das Datenbanksystem wählen unter dem man das System entwickelt. Durch bestätigen der OK Taste wird das physische Modell generiert.

Abbildung 5.9: Einstelloption beim Generieren des physischen Modells

Die Abbildung 5.8 zeigt das fertige physische Datenbankmodell. Durch die Anwendung des Regelwerks wird aus den Entities 3dmodelle und positionundrotation jeweils eine Tabelle. Die „1:n“ Beziehung zwischen den beiden Entities wird im physischen Modell dadurch erreicht, dass auf der n-Seite ein Fremdschlüssel zu jedem Datensatz eingetragen wird. Dadurch kann jede Position einem 3D Modell eindeutig zugeordnet werden. Bei Bedarf generiert Powerdesiger das passende SQL-Skript zur Erstellung der Tabellen. SQL steht für „Structured Query Language“ (strukturierte Abfragesprache). Diese Datenbanksprache enthält nicht nur Funktionen für Datenbankabfragen sondern deckt das volle Aufgabenspektrum von Anlegen und Pflegen von Datenstrukturen über das Einfügen, Ändern und Löschen von Informationen bis hin zur Benutzerverwaltung. Einen Ausschnitt aus dem SQL Skript sieht man in der Abbildung 5.10.

```

create table 3DMODELLE
(
  INDEXDREID      int not null auto_increment,
  NAME            varchar(255) not null,
  BESCHREIBUNG   text not null,
  DATEINAME      varchar(255) not null,
  primary key (INDEXDREID)
);

```

Abbildung 5.10: SQL Anweisung für Erstellung einer Tabelle

Version 4.3.2 und das Datenbanksystem MySQL in der Version 4.0.13 installiert und konfiguriert.

Für die Erstellung der PHP Skripte wird der Texteditor UltraEdit32 verwendet. Für die Übertragung der Skripte auf das Produktivsystem wird das FTP Programm, das Bestandteil jedes Betriebssystems ist, verwendet.

Um eine Datenbankabfrage ausführen zu können, muss zuvor eine Verbindung zum Datenbanksystem aufgebaut werden.

```
// Stellt Verbindung zur Datenbankserver und wählt die Datenbank aus

<?PHP
function connect_db ()
{
    $host = "localhost";           // Servername
    $mysql_db = "Diplomarbeit";    // Name der Datenbank
    $benutzer = "test";           // Benutzerkonto
    $passwort = "test";           // Passwort

    // Herstellung der Verbindung
    $connection = mysql_connect ($host, $benutzer, $passwort) or die ("Fehler: Konnte keine
                                Verbindung zum Datenbankserver
                                aufbauen.");

    // Auswahl der Datenbank
    mysql_select_db (" $mysql_db") or die ("Fehler: Konnte die Datenbank nicht auswählen");
    return $connection;           // Zugriffsnummer
}
?>
```

Abbildung 5.11: Herstellung der Verbindung zur Datenbank mit PHP

Die Abbildung 5.11 zeigt das Skript, das mit Hilfe von SQL-Anweisungen die Verbindung zur Datenbank herstellt. Wenn das Skript erfolgreich abgearbeitet wurde, wird einem eine Zugriffsnummer zugeteilt. In unserem Fall steckt die Nummer in der Variable „connection“. Mit der Nummer kann man nun Datenabfragen ausführen. Da das System viele Skripte enthält, die einen Zugang zum Datenbankserver benötigt, empfiehlt es sich, das Zugangsskript separat abzuspeichern. Benötigt ein Skript den Zugang zur Datenbank wird es mit Hilfe einer „include“-Anweisung eingebunden. Das hat den Vorteil, dass bei einer Änderung der Zugangsdaten die Änderung nur in dem Zugangsskript erfolgen muss.

Einer der wichtigsten Abfragetypen stellt die Auswahlabfrage dar und wird deshalb genauer vorgestellt. Die Syntax für eine Auswahlabfrage lautet:

Select	<i>spalte1, spalte2, ...</i>
FROM	<i>tabellennane</i>
WHERE	<i>bedingung</i>

Abbildung 5.12: SQL Select Anweisung

In der „Select“-Klausel werden die Namen einer oder mehrerer Spalten aufgelistet, die man angezeigt haben möchte. Hinter dem Schlüsselwort „FROM“ folgt der Name der Tabelle, in der die Spalten enthalten sind. Die „WHERE“-Klausel ist optional und enthält eine Bedingung. Nur diejenigen Datensätze, die die Bedingung erfüllen, werden angezeigt. Im PHP Skript sieht es wie folgt aus:

```
<?PHP
include "fcts.inc";           // bindet das Zugangsskript ein
$connection = connect_db (); // ruft die Funktion zur Herstellung der Verbindung auf

// definiert eine Auswahlabfrage in SQL
$sql01 = "SELECT name, beschreibung, dateiname from 3dmodelle";

// übergibt die Anweisung zum Ausführen an den Datenbankserver
$ergebnis = mysql_query ($sql01,$connection);
?>
```

Abbildung 5.13: PHP Skript zum Ausführen einer SQL Select Anweisung

5.5.2 Positionswerte der 3D-Modelle

Die Gesamtszenen, die mit der Exportfunktion von 3D Studio Max nach VRML umgewandelt wurden, enthalten die Positionen der einzelnen Modelle. Diese Werte müssen in die Datenbank aufgenommen werden, damit später die Gesamtszenen mit PHP erstellt werden kann.

Damit dieser Vorgang benutzerfreundlich geschehen kann wird eine Eingabemaske vorgesehen, in der die Werte eingetragen werden. Durch eine Bestätigungstaste werden die eingegebenen Werte an ein PHP Skript übergeben. Dieses trägt mit den entsprechenden SQL Anweisungen die Positionen der Modelle in die Tabellen ein.

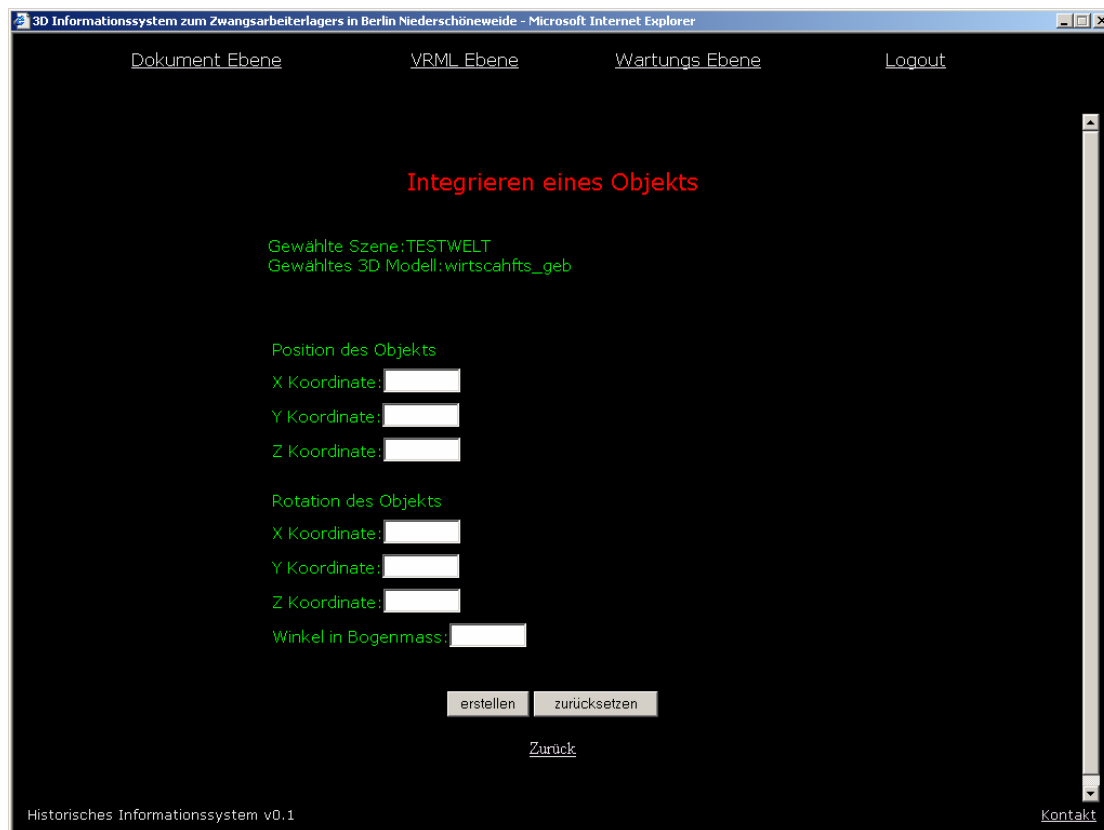


Abbildung 5.14: Eingabemaske für die Positonswerte

5.5.3 Generierung einer VRML-Datei mit PHP

Nachdem beschrieben wurde, welche Daten die Datenbank enthält, geht es nun um die Erstellung der VRML-Welt mittels der Datenbank. In VRML gibt es die Anweisung „Inline“, mit der man VRML-Objekte modular in eine Szene einbinden kann. Die Syntax für die Inline-Anweisung sieht folgendermaßen aus:

```
Inline { url „dateiname.wrl“ }
```

Abbildung 5.15: Syntax einer Inline Anweisung

Ein PHP-Skript holt sich die Positionen der jeweiligen 3D-Objekte aus der Datenbank. Anschließend können mit der Inline-Anweisung und einem Transform Knoten die Objekte in der Welt platziert werden. Der Vorteil dieser modularen Aufbauweise ist, dass die Welt leicht erweiterbar ist. Einfügen von

neuen Modellen oder Entfernen eines Modells ist somit leicht möglich. Die Abbildung 5.17 zeigt ein Beispiel mit einem PHP-Skript.

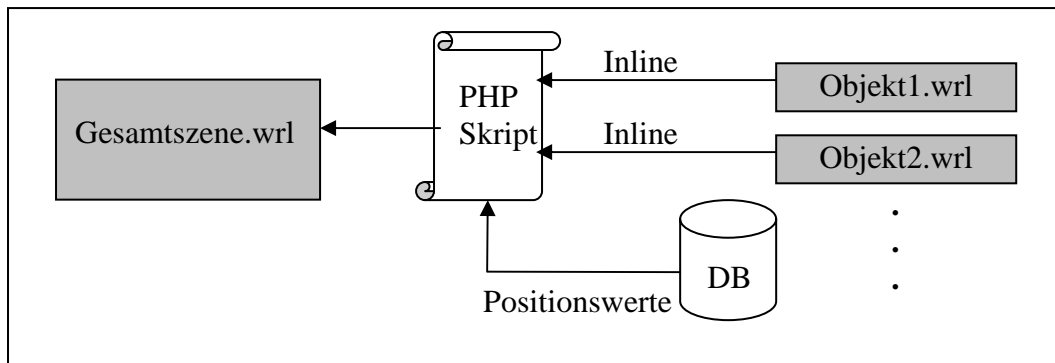


Abbildung 5.16: Erstellung einer VRML Datei grafisch dargestellt

```

<?php
Header ("Content-Type:x-world/x-vrml"); // Angabe des MIME Typs
echo "#VRML V2.0 utf8 \n"; // VRML Code entspricht den VRML 2.0 Format
echo "DEF baracke_35meter01 Transform { \n";
echo "children Inline { \n";
echo "url \"35meterzusammen1.wrl\" \n"; // Objekt in die Welt einbinden
echo "} \n";
echo "translation 9492 59.06 -13.39 \n"; // Position des Objekts
echo "rotation 0 0 1 1.571 \n"; // Objekt wird um Z Achse gedreht
echo "} \n";
?>

```

Abbildung 5.17: Erstellung einer VRML Datei mit PHP

Bei der Generierung einer VRML-Datei durch ein PHP-Skript muss man darauf achten, dass man den korrekten MIME-Typ angibt. MIME bedeutet Multipurpose Internet Mail Extension. Beim Übertragen der Daten vom Server zum Browser hat MIME die Aufgabe, dem Browser mitzuteilen, um welche Art von Daten es sich handelt und gegebenenfalls die notwendigen Plugins zu starten. Außerdem muss noch angegeben werden, welche VRML-Version und welcher Zeichensatz verwendet wird.

5.6 Informationspunkt

Dieser Abschnitt beschäftigt sich mit der Entwicklung einer Methode, um historische Dokumente zeitgleich zur VRML Welt anzuzeigen.

5.6.1 Viewpoint als Infopunkt

Damit der Benutzer alle interessanten Orte der 3D Umgebung zu sehen bekommt, gibt es bei VRML die Viewpoints. Viewpoints sind vordefinierte Kameraeinstellungen in der 3D-Welt und können je nach verwendeten VRML-Plugin über einen Schalter oder über das Steuer Menü ausgewählt werden. Die Syntax eines Viewpoints sieht wie folgt aus.

```
Viewpoint {  
  position 1 1 1      # X, Y, Z Koordinaten  
  orientation 0 0 1 0 # Drehachse ist Z und der vierte Wert gibt den Winkel an  
  description "name"  
}
```

Abbildung 5.18: Syntax der Viewpoint Anweisung

Ein Viewpoint kann in Verbindung mit einem Proximitysensor als standortabhängiger Informationspunkt genutzt werden. Ein Proximitysensor erkennt, wann ein Benutzer beim Navigieren eine bestimmte Region in der virtuellen Welt betritt. Die Region wird durch einen unsichtbaren Quader definiert, der durch die Angaben der Breite, Höhe, Tiefe und deren Mittelpunkt aufgespannt wird. Beim Betreten der Region erzeugt der Sensor ein Ereignis (isActive=TRUE) beim Verlassen ebenfalls (isActive=FALSE).

```
ProximitySensor {  
  center 1 1 1 #Mittelpunkt  
  size 1 2 3   #Ausdehnung Breite, Höhe, Tiefe  
}
```

Abbildung 5.19: Syntax der ProximitySensor Anweisung

Wählt der Besucher einen Viewpoint aus, wandert die Kamera zu den festgelegten Koordinaten. Der Proximitysensor registriert, dass die Kamera in den definierten Bereich eintritt und erzeugt ein Ereignis. Mit der Route-

Anweisung wird das Ereignis an einen Skriptknoten weitergeleitet, ausgewertet und ein PHP Skript auf dem Server aktiviert. Das PHP-Skript holt sich dann die passende Information aus der Datenbank. Textinformationen und Bilder werden jeweils in einem separaten Fenster visualisiert. Bei Bildern werden erst kleine Vorschaubilder angezeigt, um die Ladezeiten zu verringern. Durch Klicken auf ein Vorschaubild wird ein extra Fenster geöffnet, in dem das Bild in seiner Originalgröße dargestellt wird. Eine zusätzliche Textzeile gibt Auskunft darüber, woher das Material stammt.

```
#VRML V2.0 utf8

Viewpoint {
  position 1 2 3
  orientation 0 0 1 1
  description "Blick von oben"
}

DEF betreten ProximitySensor {
  center 1 2 3 #Mittelpunkt des Quaders, gleiche Position wie Viewpoint
  size 1 1 1 } #Breite, Höhe und Tiefe des Quaders

DEF holeInfo_skript Script {
  eventIn SFBool holeInfo
  field MFString index1 ["lade_doku.php"] #Name des PHP Skripts

  url "javascript:
function holeInfo (eventValue) {
  if (eventValue==TRUE) { #überprüft, ob "isActive" TRUE ist
  Browser.loadURL(index1);} #Lädt das PHP Skript
  " }

#leite das Ereignis zum Skriptknoten holeInfo,
#wenn Kamera in den definierten Bereich gelangt
ROUTE betreten.isActive TO holeInfo_skript.holeInfo
```

Abbildung 5.20: Beispiel, wie man ein Infopunkt mit VRML realisiert

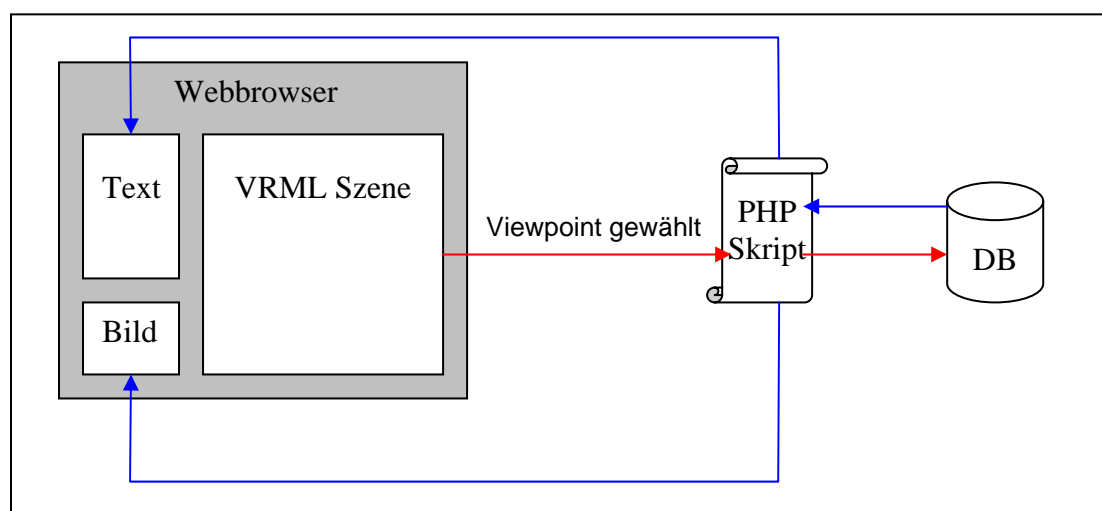


Abbildung 5.21: Infopunkt grafisch dargestellt

5.6.2 Einfügen eines Infopunkts

Autorisierte Personen sollten die Möglichkeit haben, im laufenden Betrieb einen weiteren Viewpoint in der 3D Welt zu setzen. Dieser Vorgang sollte unkompliziert und benutzerfreundlich sein. Der Nutzer sucht sich die gewünschte Welt aus und betritt sie. Mit der Maus oder der Tastatur stellt der Benutzer die gewünschte Kamerasicht ein. In dem Einfügemodus gibt es in der Welt einen Button (siehe Abbildung 5.22). Durch Drücken des Buttons wird die aktuelle Position und Rotation der Kameraeinstellung an ein PHP-Skript übergeben, wodurch sie schließlich in die entsprechende Tabelle der Datenbank eingetragen werden.

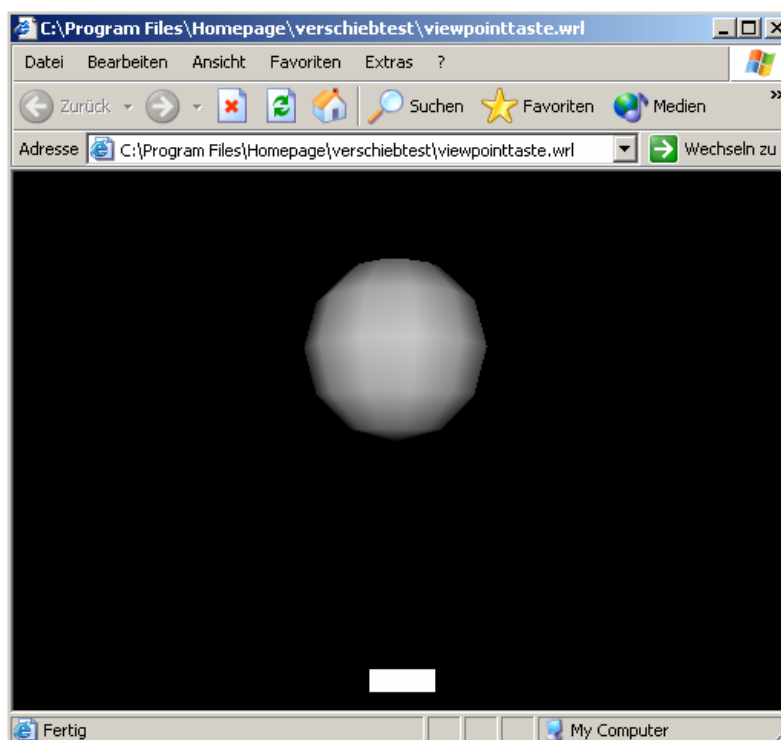


Abbildung 5.22: Button mit einer Kugel

Um das zu realisieren, wird der Proximitysensor eingesetzt. Der Proximitysensor registriert nicht nur das Betreten eines definierten Bereiches, es kann sogar jede Bewegung der Kamera erfassen. Vorausgesetzt man befindet sich im Bereich des Sensors, liefert dieser je nach Bewegung der Kamera mit dem das Ereignis „position_changed“ die neue Position und mit „orientation_changed“ die Rotation. Diese Ereignisse werden mit einer Route-Anweisung an einen Skriptknoten geschickt, wo die Werte in Variablen gespeichert werden und bei Bedarf in die Datenbank eingetragen werden. Der Button wird in VRML mit einem Quader in Verbindung mit einem Touchsensor simuliert. Der Touchsensor erzeugt ein Ereignis, wenn ein Nutzer auf ein Objekt zeigt (isOver=TRUE) oder das Objekt angeklickt hat

(isActive=TRUE). Wird der Button gedrückt, aktiviert das Ereignis einen Skriptknoten, der die gespeicherten Werte über den Hyperlink an ein PHP Skript übergibt.

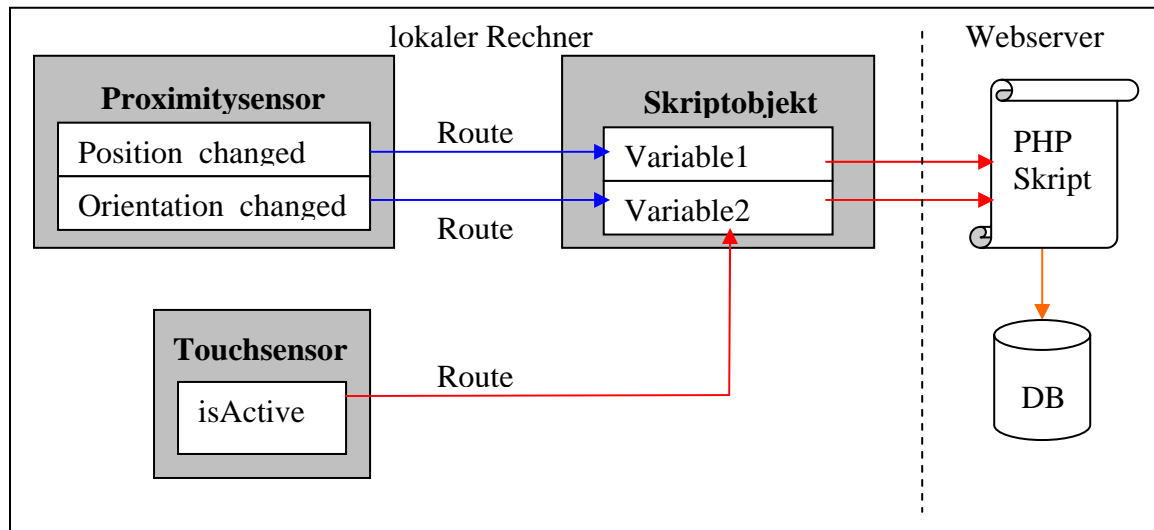


Abbildung 5.23: Ablauf für Einfügen eines Infopunkts grafisch dargestellt

Alle 3D-Objekte in der VRML-Welt sind positioniert durch ihre Koordinaten. Würde sich die Kamera um 90 Grad nach links bewegen, so würde der Button aus dem Blickwinkel verschwinden. Damit das nicht geschieht, müssen die Positionsänderungen dem Button mitgeteilt werden, sodass er seine Position zur Kamera anpassen kann. Indem man die Ereignisse „position_chahnged“ und „orientation_chahnged“ mit einer Route-Anweisung an das Button Objekt koppelt, wird der Button mit Hilfe der Anweisungen „set_translation“ und „set_rotation“ zu den Koordinaten verschoben.

Damit der Sensorbereich des Proximitysensors nicht unnötig groß definiert werden muss, wird die Position des Sensors ebenfalls relativ zur Position der Kamera angepasst.

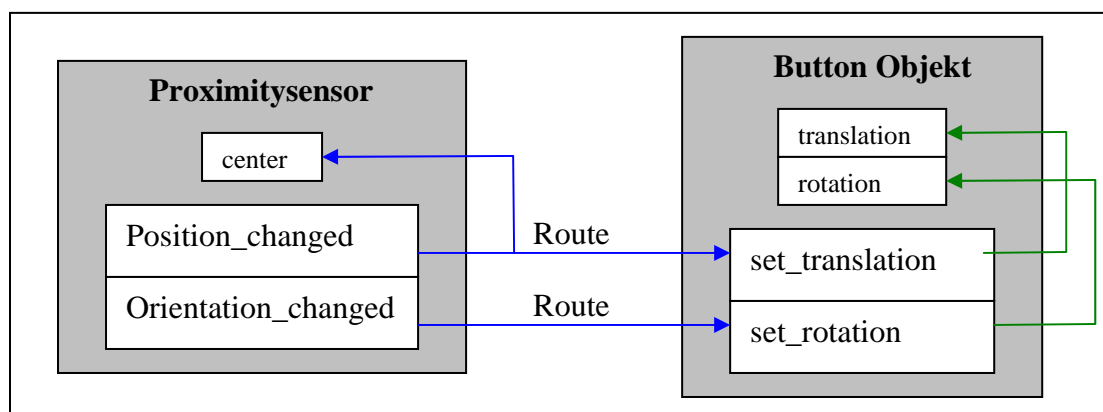


Abbildung 5.24: Methode damit Button Objekt stets im Blickfeld ist

5.7 Bewegen der 3D-Objekte

Bei historischen Rekonstruktionen kann es passieren, dass durch ungenaue Angaben der Quellen die Lage der 3D Objekte nachträglich korrigiert werden muss. Deshalb soll das System Positionsänderungen ermöglichen. Aus Gründen der Benutzerfreundlichkeit wird das Verschieben und Rotieren der 3D-Objekte nicht durch Eingabe von Werten vorgenommen, sondern durch eine Steuerkonsole. Dieser Abschnitt beschreibt, welche Funktionen die Steuerkonsole besitzen muss und wie die Funktionen realisiert werden.

5.7.1 Auswahlverfahren des 3D-Objekts

Bevor eine Manipulation am Objekt gemacht werden kann, muss das Objekt erst ausgewählt werden. Die einfachste Variante wäre eine Liste auszugeben, in der alle Objekte der Szene verzeichnet sind. Durch Markieren des gewünschten Objekts würde die Auswahl getroffen. Diese Methode hat den Nachteil, dass bei einer Szene mit vielen Objekten die Liste schnell lang wird und man die Übersicht verliert.

Besser ist es, wenn die Auswahl in der VRML-Welt geschieht. Der Nutzer betritt die Szene und bewegt sich zu dem Objekt, das er gerne ändern möchte. Mit dem Mauszeiger klickt er das Objekt an. Ein Skriptknoten registriert das und aktualisiert die Szene. Nach dem Aktualisierungsvorgang wird die Steuerkonsole eingeblendet mit der man das gewählte Objekt verschieben bzw. rotieren kann.

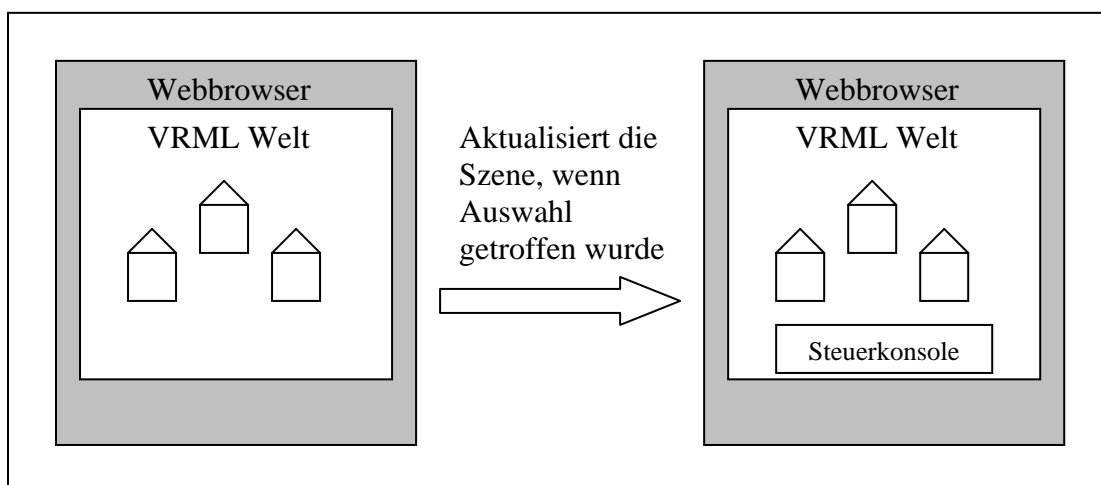


Abbildung 5.25: Auswahl des 3D Objekts

5.7.2 Steuerkonsole

Die Position eines Objektes im dreidimensionalen Raum wird durch die Angabe der X-, Y- und Z-Koordinate bestimmt. Will man das Objekt auf der Horizontalen nach rechts verschieben, so muss der X-Wert erhöht werden. Reduziert man den Wert, so verschiebt sich das Objekt nach links. Daraus ergibt sich, dass die Steuerkonsole mindestens sechs Buttons benötigt: Je Koordinate zwei Buttons, die den Wert erhöhen bzw. den Wert reduzieren. Für die Rotation um die X-, Y- und Z-Achse würden eigentlich noch mal sechs Buttons benötigt. Aus Gründen der Übersichtlichkeit werden die vorhandenen sechs Buttons mit einer Doppelfunktion belegt. Über zwei zusätzliche Schalter kann zwischen den Funktionen Rotieren und Verschieben gewechselt werden. Die Abbildung 5.26 zeigt wie die Steuerkonsole aussehen muss.

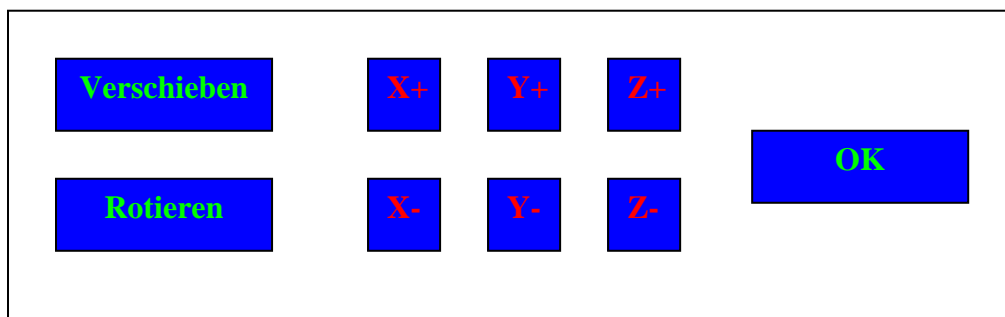


Abbildung 5.26: Steuerkonsole

Ein OK-Button trägt schließlich die Änderungen in die Datenbank ein.

Da nun alle wichtigen Buttons für die Steuerkonsole definiert sind, wird im Folgenden die Implementierung beschrieben.

In VRML kümmert sich der Transform Knoten um die Positionierung und Rotation der Objekte. Die Abbildung 5.27 zeigt den VRML-Kode mit der ein Würfel in die Position 0 1 2 gebracht wird und eine 90 Grad Rotation um die Z-Achse gemacht wird.

```
#VRML V2.0 utf8
Transform {
  Children [
    geometry   Shape {
      Box { 2 2 2 } #Erzeugt ein Würfel
    }
  ]
  translation 0 1 2 #Verschiebung der Kugel
  rotation    0 0 1 1.5707 #90° Rotation um die Z Achse
}
```

Abbildung 5.27: Beispiel für Transform Anweisung

Der Transform Knoten besitzt zwei Felder: das „translation“ und das „rotation“ Feld. Mit den drei Werten X, Y, Z wird beim Feld „translation“ die Lage des Objektes bestimmt. Rotation besitzt hingegen vier Werte. Die ersten drei Komponenten geben die Drehachse an, die vierte den Winkel im Bogenmaß. Wird zum Beispiel um die X-Achse gedreht, wird die Kombination 1 0 0 benötigt, bei der Y-Achse 0 1 0 und bei der Z-Achse 0 0 1. Beim Wechseln der Rotationsachse beginnt die Drehung immer von der Ausgangsposition.

Die Translations- und Rotationswerten des ausgewählten 3D-Objektes werden als Variablen in einem Skriptknoten gespeichert. Drückt der Nutzer den x+ Button so empfängt ein Skriptknoten das Ereignis. Die im Skript gespeicherten Positionswerte werden entweder im Verschieben Modus oder Rotieren Modus entsprechend erhöht. Die neuen Werte werden mit der Route Anweisung an den Würfel geschickt und mit dem Befehl „set_translation“ und „set_rotation“ wird das Objekt zu den neuen Koordinaten verschoben. Beim Drücken des „OK“ Buttons werden die gespeicherten Werte des Skriptsknotens über ein Hyperlink an ein PHP Skript übergeben und in die Datenbank eingetragen.

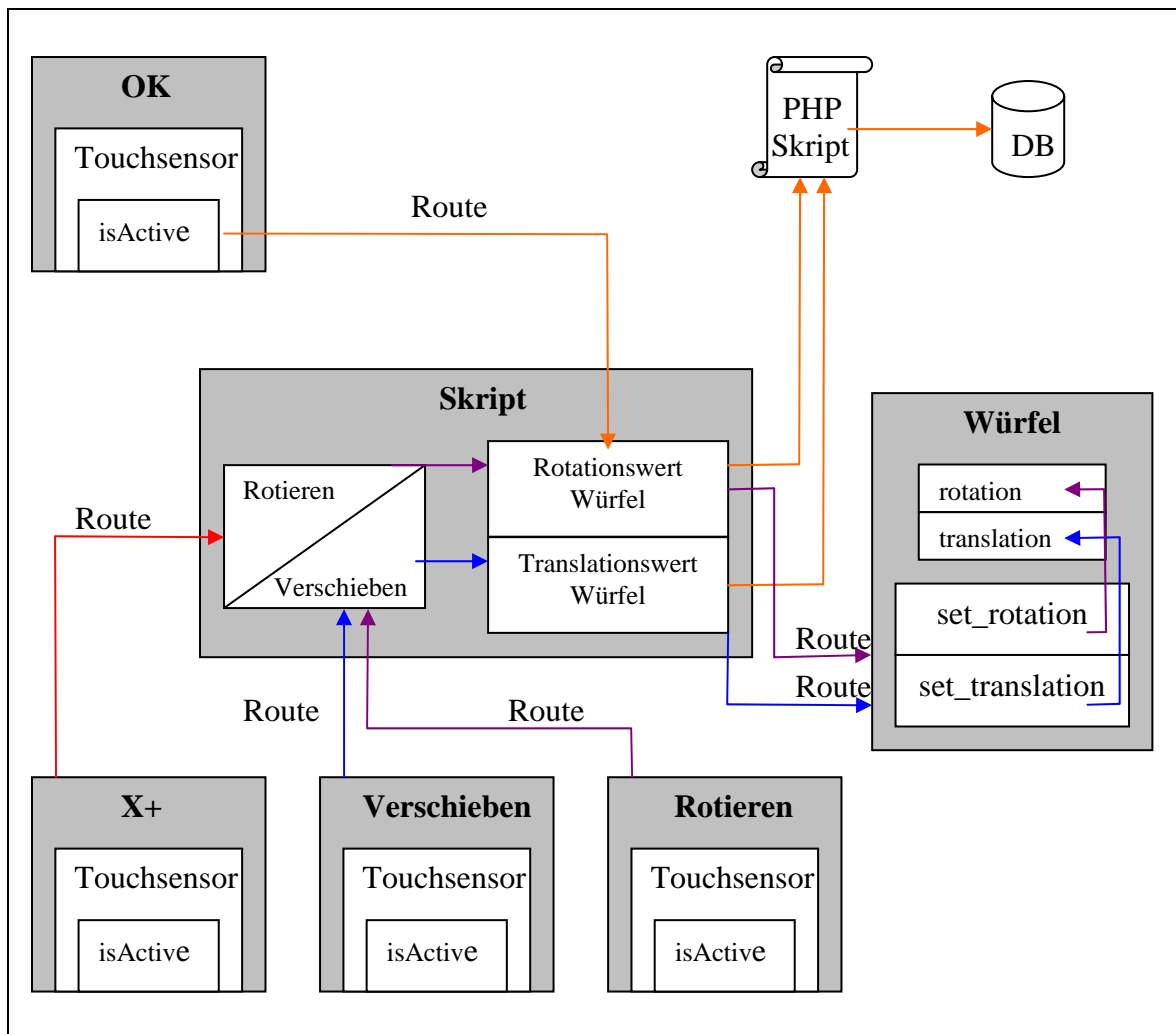


Abbildung 5.28: Interaktives Verschieben grafisch dargestellt

5.8 Webinterface zur Verwaltung des Systems

Will man ein neues Dokument auf den Server Hochladen, muss im Normalfall ein FTP-Programm benutzt werden. Änderungen an Tabellen müssen über SQL-Anweisungen erfolgen. Für einen Verwalter des Systems bedeutet das ein nicht zu unterschätzenden Aufwand, da er Kenntnisse der Datenbanksprache SQL benötigt und den Umgang mit einem FTP Programm beherrschen muss.

Viel einfacher wäre es, wenn alle Operationen über eine einheitliche Benutzeroberfläche erfolgen würde. Einfügen oder Ändern von Tabelleneinträgen oder Hochladen von Dateien auf dem Server sollten durch Mausklicks erledigt werden können.

Ein Webinterface kann hier helfen, alle notwendigen Prozeduren, die für die Verwaltung des Systems notwendig sind, über eine grafische Oberfläche auszuführen.

Der Vorteil der sich daraus ergibt ist, dass der Verwalter oder jede beliebige autorisierte Person durch kurze Einarbeitungszeit in der Lage ist, neue 3D-Modelle Hochzuladen und in eine neue oder bestehende Welt zu integrieren. Neue Dokumente können in wenigen Schritten einem Informationspunkt zugeordnet werden. Ein Webinterface bietet den Vorteil, dass man das System von jedem Computer mit einem Internetanschluss und einem Webbrowser aus verwalten kann.

Für die Erstellung des Webinterfaces wird neben der HTML-Technologie PHP und Cascading Stylesheets kurz CSS eingesetzt. Mit PHP werden SQL-Anweisung ausgeführt und Funktionen für den Dateiupload bereitgestellt. HTML und CSS werden für das Layout der Seiten und die Visualisierung der Informationen genutzt. Die Abbildung 5.29 zeigt die Oberfläche für den Dateiupload. Mit dem „Durchsuchen“-Button wird die Datei gewählt, die hochgeladen werden soll. Mit den beiden anderen Eingabefeldern kann der Datei eine Beschreibung und ein Name vergeben werden. Durch betätigen des Hochladen-Buttons wird die Datei übertragen. Ein PHP-Skript überwacht den Vorgang und meldet ob der Vorgang erfolgreich war oder nicht.

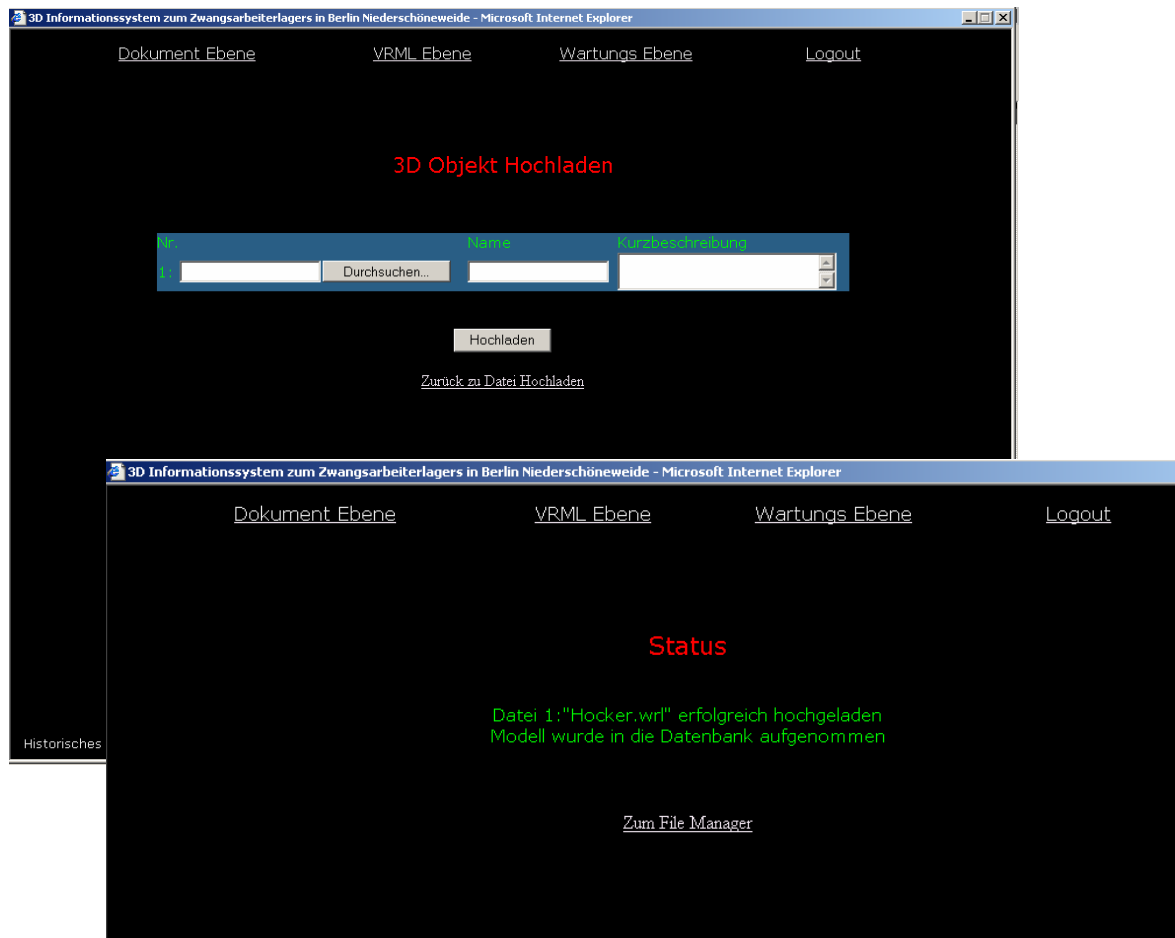


Abbildung 5.29: Webinterface des Systems

6. Ergebnisse

6.1 Bedienung des Prototyps

Das System gliedert sich in drei Hauptebenen. Die Dokumentebene, die Vrmlebene und die Wartungsebene.

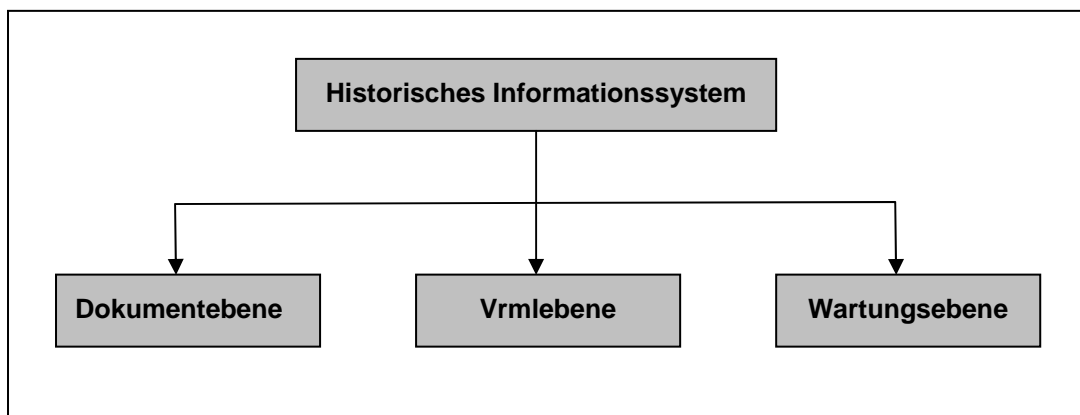


Abbildung 6.1: Menü des Systems

Diese Ebenen können jeder Zeit über das Menü, das sich im oberen Bereich des Browserfenster befindet, ausgewählt werden. Alle nicht notwendigen Anzeigeelemente des Browser sind deaktiviert, um eine möglichst große Anzeigefläche zu erhalten. Auf die Dokumentebene und die Vrmlebene hat jeder Benutzer Zugriff. Die Wartungsebene ist nur für den Systemverwalter gedacht und erfordert eine Autorisation des Nutzers. Nach erfolgreichem Anmelden wird das Menü um den Unterpunkt Wartungsebene erweitert. Des Weiteren wird noch eine Abmeldefunktion zur Verfügung gestellt.



Abbildung 6.2: Einstiegspunkt des Systems

Auf der Vrml-Ebene kann sich der Benutzer eine der drei virtuellen Welten aussuchen. Nach der Auswahl wird die VRML-Welt in ein Frame geladen. Damit der Nutzer stets weiß, in welcher virtuellen Welt er sich gerade befindet, gibt ein zusätzliches Frame den Namen der virtuellen Welt an. Die Darstellung der standortabhängigen Informationen werden über zwei zusätzliche Frames angezeigt. Der eine Frame blendet Information in Form von Text ein und der andere Frame die Bilder. Damit das Herunterladen der Bilder möglichst schnell geht, werden Vorschaubilder in der Größe von 80x80 Pixeln angeboten (siehe Abbildung 6.3).

Durch Klicken auf ein Vorschaubild öffnet ein Javascript ein Browserfenster, in der das Bild in seiner Originalgröße dargestellt wird (siehe Abbildung 6.4). Dadurch wird der Datentransfer auf ein Minimum reduziert was bei einer langsamen Internetverbindung Vorteilhaft ist. Beide Frames werden jedes Mal aktualisiert wenn ein Viewpoint angewählt wurde. Das Navigieren in der virtuellen Welt erfolgt mit der Tastatur oder mit der Computermaus.



Abbildung 6.3: Geländeszene mit Infopunkt



Abbildung 6.4: Geländeszene mit Archivbilder

Damit der Nutzer möglichst schnell auf alle Informationen zugreifen kann, wird eine Dokumentebene angeboten.

Die Dokumentebene umfasst alle Informationen, die in den drei VRML-Welten verwendet werden. Auf der Dokumentebene kann der Nutzer die Annotationen und die dazugehörigen Bilder abrufen. Bei Bedarf führt ein zusätzlicher Link den Nutzer zum Standort in der virtuellen Welt, wo diese Materialien verwendet werden.

Ein Bilderarchiv, das alphabetisch sortiert ist, gibt einen Überblick über alle Bilder. Um lange Ladezeiten zu vermeiden werden erst Vorschaubilder angezeigt.

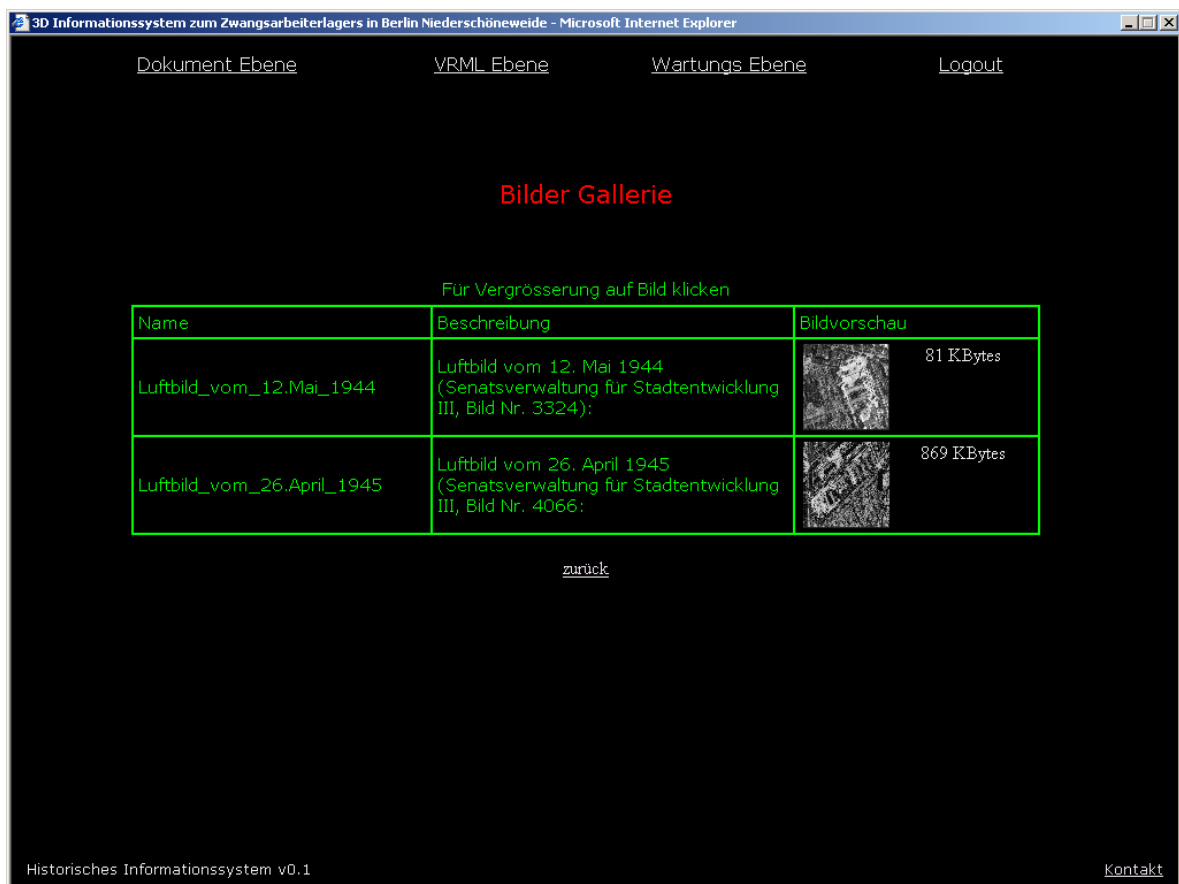


Abbildung 6.5: Bildergalerie

Die Wartungsebene ist in vier Untermenüs unterteilt.

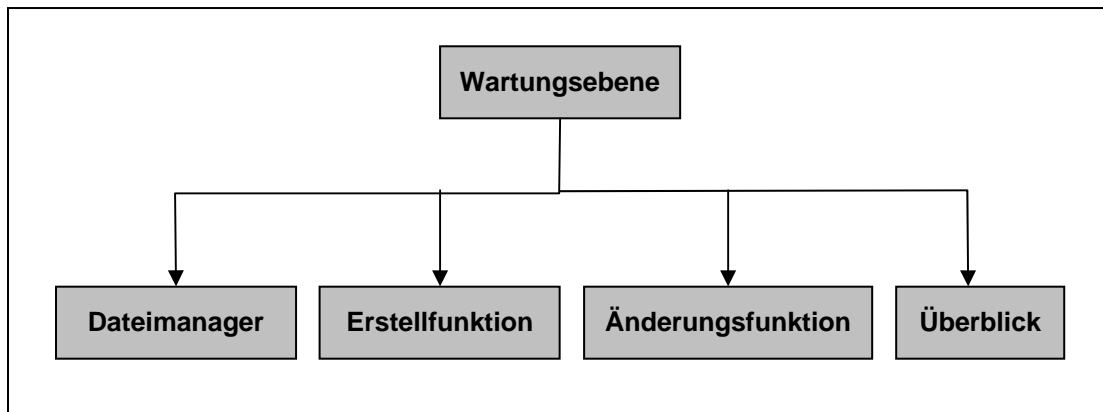


Abbildung 6.6: Untermenü von Wartungsebene

Der Dateimanager dient dazu, 3D-Objekte und Bilder zum Server zu übertragen. Beim Übertragen eines Bildes wird gleichzeitig ein Vorschaubild erzeugt und in ein Verzeichnis „Vorschaubilder“ abgelegt. Dateien, die nicht mehr benötigt werden, können ebenfalls über den Dateimanager gelöscht werden. Beim Löschen eines Bildes wird automatisch das dazugehörige Vorschaubild mit entfernt.

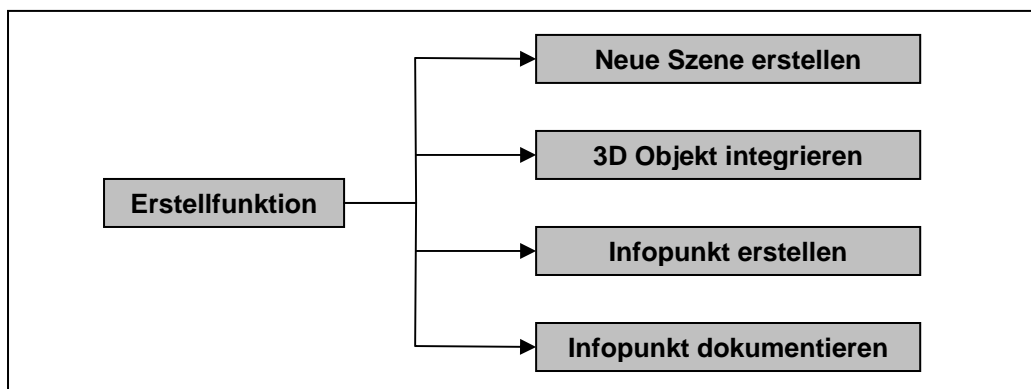


Abbildung 6.7: Gliederung von Erstelfunktion

Das Untermenü Erstelfunktion bietet Funktionen zum Erstellen von neuen Szenen an. Dadurch kann das System durch weitere VRML-Welten erweitert werden. Mit Hilfe des Menüpunkts „3D Objekte integrieren“ können neue 3D-Objekte in eine bestehende oder neue Welt hinzugefügt werden. Durch die Auswahl der Szene und des neuen 3D-Objektes wird mit den Eingaben der Positionswerte das 3D-Objekt in die vorgesehene Welt eingebunden. Das Einfügen eines neuen Viewpoints, der als standortabhängiger Infopunkt dient

(siehe Kapitel 5.6), erfolgt mit dem Menüpunkt „Infopunkt erstellen“. Das Verfassen einer Annotation bzw. das Anhängen von Bildern an einen Infopunkt kann mit „Infopunkt dokumentieren“ erfolgen.

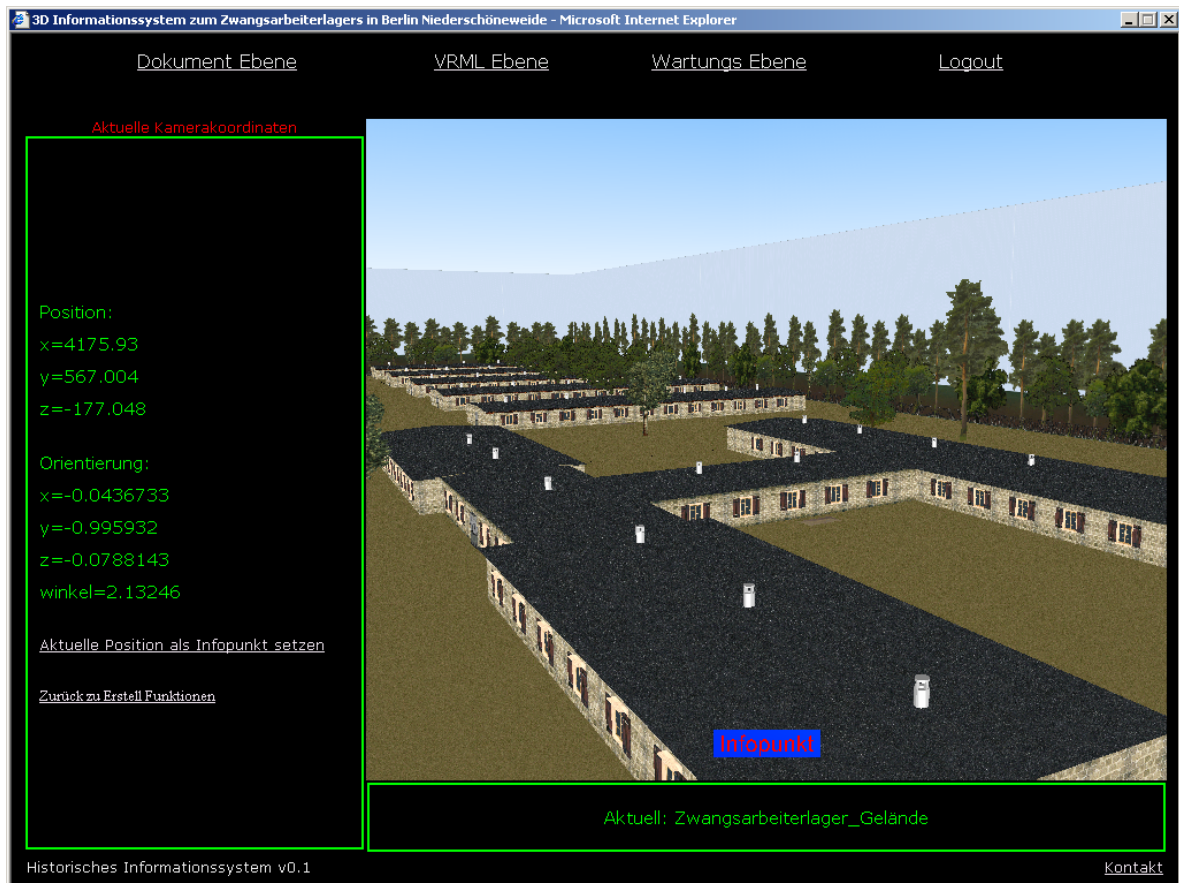


Abbildung 6.8: Infopunkt setzen

Um nachträglich Änderungen vornehmen zu können gibt es den Menüpunkt Änderungsfunktion. Der Name der Szene kann geändert werden oder die komplette Szene entfernt werden. Das interaktive Verschieben eines 3D Modells (siehe Kapitel 5.7) sowie das Entfernen eines 3D Objektes aus einer Welt kann von hieraus erfolgen. Funktionen zur Bearbeitung und Löschen eines Viewpoints sind ebenfalls vorhanden. Das Login Passwort kann nach Wunsch geändert werden.

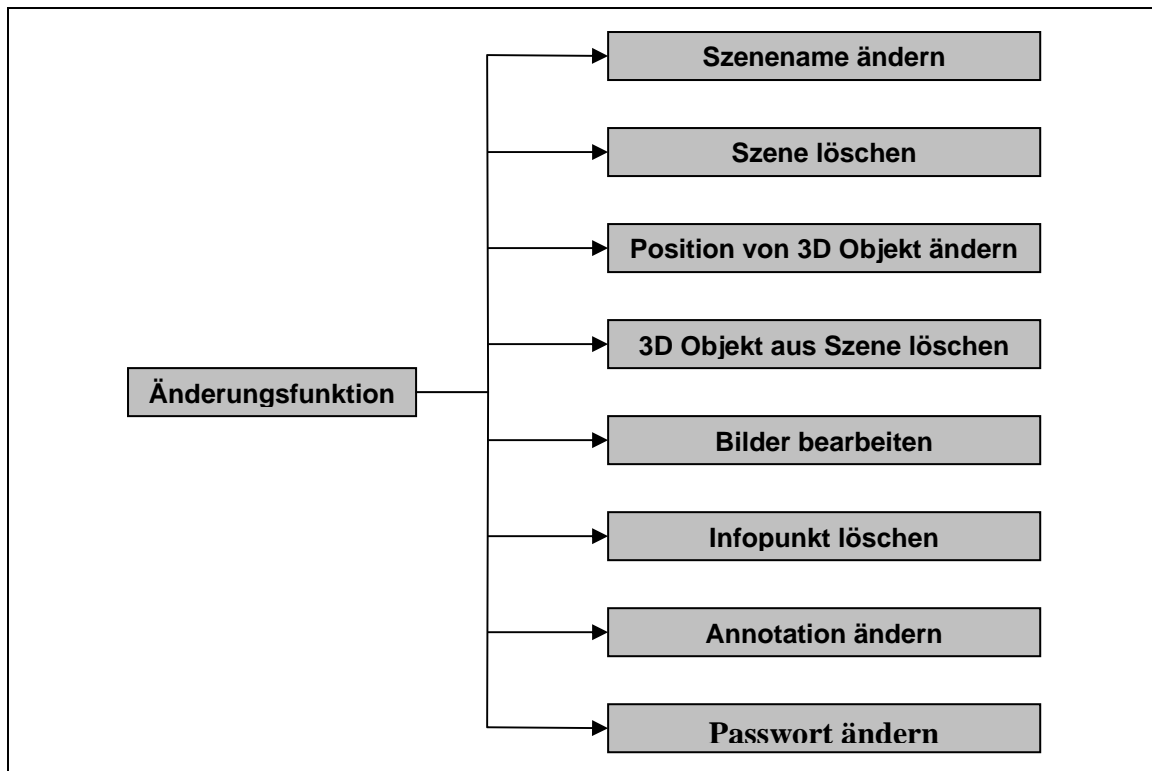


Abbildung 6.9: Gliederung von Änderungsfunktion

Der Menüpunkt Überblick gibt dem Systemverwalter darüber Auskunft, wieviele 3D-Objekte das System verwaltet und in welcher Szene sie verwendet werden. Durch Selektieren einer Szene werden alle verwendeten 3D Modelle und eingerichteten Infopunkte der virtuellen Welt tabellarisch aufgelistet. 3D-Modelle können außerdem separat angeschaut und der dazugehörige VRML Quellcode in einen extra Fenster eingeblendet werden.

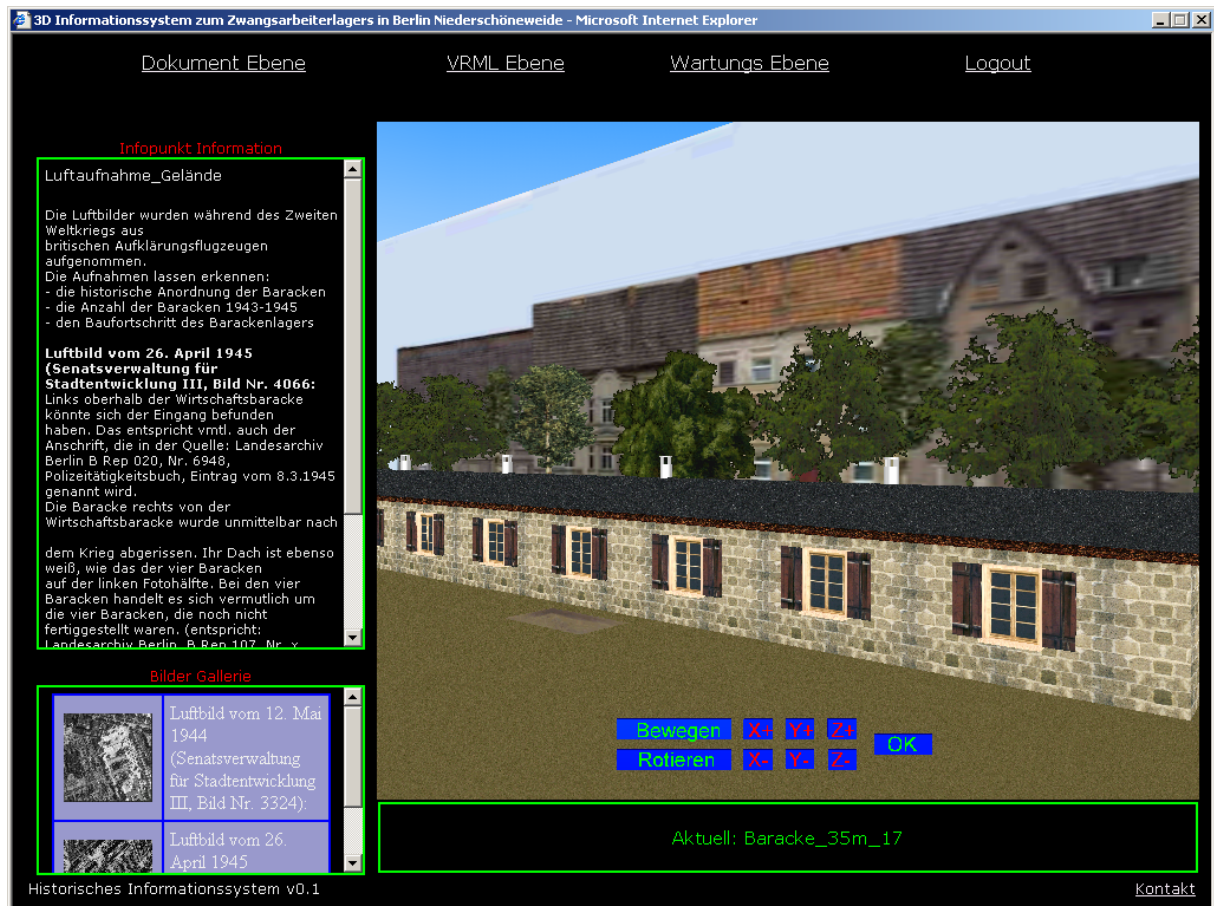


Abbildung 6.10: Geländeszene mit Steuerkonsole

6.2 Performance Test

Die FHTW Berlin bietet für jeden Studenten einen Webspaces auf ihrem Webserver ein. Für Testzwecke wird der Prototyp auf dem Server eingerichtet, da dieser die Anforderungen der Arbeit erfüllt. Der Webserver läuft mit der Apache Software und unterstützt die Skriptsprache PHP. Außerdem wird ein Zugriff auf eine MySQL Datenbank angeboten.

Mit der URL <http://home.fhtw-berlin.de/~s0234194/index.php> lässt sich der Prototyp aufrufen. Nach erfolgreichen Testverläufen kann das System z.B. in die Webseite der Geschichtswerkstatt eingebunden werden.

Der Prototyp wird mit dem Internetbrowser Internet Explorer 6 und mit dem VRML Plugin Blaxxun bei einer Bildschirmauflösung von 1024x768 getestet. Als Testplattform wird ein Computer mit einem AMD XP1700+ CPU und 512 MB RAM Hauptspeicher verwendet. Eine Grafikkarte von der Firma nVidia mit dem Geforce2 Titanium Chip und 64 MB Videospeicher sorgt für die Visualisierung der VRML-Welten.

Um zu erfahren, wie lange es dauert bis eine komplette VRML-Welt vom Webserver auf dem lokalen PC übertragen ist, wird die Zeit gemessen.

Als Internetverbindung steht ein ADSL Anschluss mit einer Datentransferrate von 768kbit/s zu Verfügung. Da nicht jeder Nutzer einen Breitbandanschluss hat, wird noch die Zeit ermittelt, die ein ISDN-Anschluss braucht. Zum Testzeitpunkt lag kein ISDN-Anschluss bereit. Deshalb wird die Zeit rechnerisch ermittelt. Ein ISDN-Anschluss hat eine Übertragungsrate von 6,4kbit/s und ist somit 12mal langsamer als der zu Verfügung stehende ADSL Anschluss. Die gemessene Zeit wird mit 12 bzw. mit 6 bei Kanalbündelung multipliziert.

Anschlusstyp	Szene1	Szene2	Szene3
ADSL	25 Sek.	22 Sek.	12 Sek.
ISDN Kanalbündelung	150 Sek.*	132 Sek.*	72 Sek.*
ISDN einfach	300 Sek.*	264 Sek.*	144 Sek.*

Tabelle 7.1: Zeitübersicht

Szene1 = Gelände

Szene2 = Baracke von Innen

Szene3 = Luftschutzkeller

* rechnerisch ermittelt

Die gemessenen Ladezeiten zeigen, dass zum Betrachten der VRML Szenen ein schneller Internetanschluss benötigt wird. Der Einsatz eines ISDN Anschlusses führt zu extrem langen Wartezeiten. Auch bei einer ISDN Kanalbündelung, der die Zeit um die Hälfte reduziert, muss der Nutzer mit langen Wartezeiten rechnen.

Eine Marktforschung [38] zeigt, dass immer mehr Internetnutzer sich für einen Breitbandinternetanschluss entscheiden. Da die Zahl Jahr für Jahr steigt, sollte das Problem mit den langen Ladezeiten in Zukunft kein Problem mehr darstellen.

7 Ausblick

Als Ergebnis ist ein dreidimensionales historisches Informationssystem entstanden. Das System bietet neben der 3D-Rekonstruktion des Geländes eine ausführliche Dokumentation zur Geschichte des Zwangsarbeiterlagers in Niederschöneweide. Der Nutzer kann sich in den virtuellen Welten frei bewegen und in der jeweiligen Welt über Infopunkte die dazugehörige Dokumentation abrufen. Die Dokumentation besteht aus Bildern und Texten und kann in Verbindung mit der VRML-Welt oder separat angeschaut werden.

Das System ist so entwickelt, dass es durch weitere 3D-Welten erweiterbar ist. Mit dem Webinterface lassen sich die neuen 3D-Modelle zum Server übertragen und in die neue Welt integrieren. Außerdem bietet das Interface Funktionen zur Verwaltung der Dokumente und der virtuellen Welten an.

Folgende Erweiterungen wären denkbar. Um ein Feedback vom Nutzer zu erhalten, ist das Einrichten eines Diskussionsforums oder ein Gästebuch sinnvoll. Dort können Meinungen geäußert werden wie z.B. ein interaktives 3D-Modell für die pädagogische Arbeit einsetzbar ist oder ob die virtuelle Rekonstruktion einen realen Rückbau ersetzen kann. Fehler bei der Rekonstruktion eines Gebäudes oder Verbesserungsvorschläge um das System benutzerfreundlicher zugestalten können gemeldet werden.

Die Berliner Geschichtswerkstatt bietet nur für spezielle Anlässe Rundgänge auf dem Zwangsarbeitergelände an. Damit der Nutzer auch außerhalb dieser speziellen Anlässe das Gelände besichtigen kann, könnte die Quicktime [35] VR Technologie der Firma Apple eingesetzt werden, mit der man 360 Grad Panoramenbilder erstellen kann. Dadurch können Gelände und die Innenräume der Gebäude des Zwangsarbeiterlagers in ihrem heutigen Zustand photorealistisch abgebildet werden.

Das Einfügen eines 3D Modells in eine virtuelle Welt erfolgt zur Zeit durch Eingabe der Positionswerte. Die Positionswerte, die in der Gesamtszene einer VRML-Datei enthalten sind, müssen manuell abgelesen werden. Da dies mit viel Arbeit verbunden ist, sollte eine Alternative gefunden werden, wie die Objekte ohne viel Aufwand eingebunden werden können. Eine denkbare Lösung

wäre, wenn alle neuen 3D Objekte zu einem festgelegten Ort der Welt eingefügt werden. Der Ort kann z .B. noch mit einem roten Kreis verdeutlicht werden. Anschließend wird das 3D-Objekt mit Hilfe der Steuerkonsole (siehe Kapitel 5.7) zu der passenden Position hingeschoben und mit dem OK Button die Positionswerte in der Datenbank eingetragen.

Abbildungsverzeichnis

Abbildung 1.1: Homepage der Berliner Geschichtswerkstatt.....	- 6 -
Abbildung 3.1: Menü von Theatron.....	- 10 -
Abbildung 3.2: Interface, Theater von Dionysos.....	- 10 -
Abbildung 3.3: Vollsicht auf das Theater von Dionysos.....	- 11 -
Abbildung 3.4: Sicht von Oben.....	- 11 -
Abbildung 3.5: Sicht von der Seite.....	- 11 -
Abbildung 3.6: Screenshots von dem System VRND.....	- 13 -
Abbildung 3.7: BBC Virtual Tours.....	- 14 -
Abbildung 3.8: Einstiegspunkt Endeavour.....	- 15 -
Abbildung 3.9: VRML Modell des Segelschiffs Endeavour.....	- 15 -
Abbildung 3.10: Einstiegspunkt St.Edmunds Abtei.....	- 16 -
Abbildung 3.11: VRML Modell der St.Edmunds Abtei.....	- 16 -
Abbildung 4.1: HMD.....	- 18 -
Abbildung 4.2: CAVE.....	- 19 -
Abbildung 4.3: Datenhandschuh.....	- 19 -
Abbildung 4.4: Benutzeroberfläche von Macromedia Director.....	- 21 -
Abbildung 4.5: Java3D Virtual Universe.....	- 25 -
Abbildung 4.6: VRML Szenengraph.....	- 29 -
Abbildung 4.7: Route Anweisung.....	- 31 -
Abbildung 4.8: VRML-Builder Cosmo World.....	- 32 -
Abbildung 4.9: VRML Code.....	- 32 -
Abbildung 4.10: Benutzeroberfläche von X3D Edit.....	- 36 -
Abbildung 4.11: X3D Quellcode.....	- 36 -
Abbildung 5.1: Benutzeroberfläche von 3D Studio Max.....	- 43 -
Abbildung 5.2: Export nach VRML.....	- 43 -
Abbildung 5.3: Exportfunktion von 3D Studio Max.....	- 44 -
Abbildung 5.4: Ergebnis des Exports mit VRML plugin betrachtet.....	- 44 -
Abbildung 5.5: VRML Code mit DEF Anweisung.....	- 45 -
Abbildung 5.6: VRML Code mit DEF und USE Anweisung.....	- 46 -
Abbildung 5.7: Entity Relation Ship Modell.....	- 48 -
Abbildung 5.8: Physisches Modell.....	- 50 -
Abbildung 5.9: Einstelloption beim Generieren des physischen Modells.....	- 49 -
Abbildung 5.10: SQL Anweisung für Erstellung einer Tabelle.....	- 49 -
Abbildung 5.11: Herstellung der Verbindung zur Datenbank mit PHP.....	- 51 -
Abbildung 5.12: SQL Select Anweisung.....	- 51 -

Abbildung 5.13: PHP Skript zum Ausführen einer SQL Select Anweisung .	- 52 -
Abbildung 5.14: Eingabemaske für die Positonswerte	- 53 -
Abbildung 5.15: Syntax einer Inline Anweisung.....	- 53 -
Abbildung 5.16: Erstellung einer VRML Datei grafisch dargestellt	- 54 -
Abbildung 5.17: Erstellung einer VRML Datei mit PHP	- 54 -
Abbildung 5.18: Syntax der Viewpoint Anweisung	- 55 -
Abbildung 5.19: Syntax der ProximitySensor Anweisung	- 55 -
Abbildung 5.20: Beispiel, wie man ein Infopunkt mit VRML realisiert.....	- 56 -
Abbildung 5.21: Infopunkt grafisch dargestellt	- 56 -
Abbildung 5.22: Button mit einer Kugel.....	- 57 -
Abbildung 5.23: Ablauf für Einfügen eines Infopunkts grafisch dargestellt. -	58 -
Abbildung 5.24: Methode damit Button Objekt stets im Blickfeld ist	- 58 -
Abbildung 5.25: Auswahl des 3D Objekts.....	- 59 -
Abbildung 5.26: Steuerkonsole	- 60 -
Abbildung 5.27: Beispiel für Transform Anweisung.....	- 60 -
Abbildung 5.28: Interaktives Verschieben grafisch dargestellt.....	- 61 -
Abbildung 5.29: Webinterface des Systems	- 63 -
Abbildung 6.1: Menü des Systems.....	- 64 -
Abbildung 6.2: Einstiegspunkt des Systems	- 65 -
Abbildung 6.3: Geländeszene mit Infopunkt	- 66 -
Abbildung 6.4: Geländeszene mit Archivbilder.....	- 66 -
Abbildung 6.5: Bildergalerie.....	- 67 -
Abbildung 6.6: Untermenü von Wartungsebene.....	- 68 -
Abbildung 6.7: Gliederung von Erstellfunktion	- 68 -
Abbildung 6.8: Infopunkt setzen.....	- 69 -
Abbildung 6.9: Gliederung von Änderungsfunktion	- 70 -
Abbildung 6.10: Geländeszene mit Steuerkonsole	- 71 -

Tabellenverzeichnis

Tabelle 3.1: Systemvergleich	- 17 -
Tabelle 4.1: Vor- und Nachteile Macromedia Director	- 22 -
Tabelle 4.2: Vor- und Nachteile von 3D-Gameengine	- 24 -
Tabelle 4.3: Vor- und Nachteile von Java3D	- 28 -
Tabelle 4.4: Vor- und Nachteile von VRML	- 34 -
Tabelle 4.5: Vor- und Nachteile von X3D	- 37 -
Tabelle 7.1: Zeitübersicht.....	- 72 -

Quellenverzeichnis

Literatur Quellen

- [1] Tobias Hager
Träume Online
iX, 05/2000

- [2] Eckhard Ammann
Programmierung animierter Welten
International Thomson Publishing, 1997

- [3] Stephan Diehl
Java & Co
Addison Wesley, 1997

- [4] Jörg Kloss; Robert Rockwell u.a.
VRML97 – Der neue Standard für interaktive 3D Welten im WWW
Addison Wesley, 1998

- [5] Hans-Lothar Hase
Dynamische virtuelle Welten mit VRML 2.0
dpunkt Verlag, 1997

- [6] Rolf Däßler; Hartmut Palm
Virtuelle Informationsräume mit VRML
dpunkt Verlag, 1998

- [7] Jason Roberts; Phil Gross
Director 8
Addison Wesley, 2000

-
- [8] Ralph Steyer
JavaScript in 21 Tagen
Markt+Technik Verlag, 2003
- [9] Thomas Kobert
Das Einsteigerseminar HTML4
bhv, 2001
- [10] Dr. Susanne Wigard
Das Einsteigerseminar PHP4
bhv, 2001
- [11] Hajo Schulz
Datenschule
c't Magazin für Computer und Technik, 9/2003, Seite 234-239
- [12] Hajo Schulz; Peter Siering
Datendiener, Freie Datenbankserver im Vergleich
c't Magazin für Computer und Technik, 5/2003, Seite 142-147
- [13] Thomas Sawatzki
Virtuelle Galerie für die Deutsche Guggenheim Berlin
Entwicklung eines webbasierten 3D-Präsentationssystems und
Verwaltung der Objekte auf MySQL Basis
Diplomarbeit FHTW-Berlin, 2002
- [14] Andreas Vogel
3D-Informationssysteme zur Visualisierung historischer
Stadtlandschaften am Beispiel vom Berlin des 18. Jahrhunderts
Diplomarbeit FHTW-Berlin, 2002

Internet Quellen

- [15] <http://www.berliner-geschichtswerkstatt.de>
10/2003, Homepage der Berliner Geschichtswerkstatt
- [16] <http://www.theatron.org>
10/2003, Theatron
- [17] <http://www.vrndproject.com>
10/2003, VRND
- [18] <http://www.idsoftware.com/>
10/2003, IDSoftware
- [19] <http://www.epicgames.com/>
10/2003, Epicgames
- [20] <http://www.valvesoftware.com/projects.htm>
10/2003, Valve
- [21] http://www.bbc.co.uk/history/multimedia_zone/virtual_tours/
10/2003, BBC Virtual Tours
- [22] <http://www.macromedia.com/software/director/>
10/2003, Macromedia Director
- [23] <http://www.datagrip.com/>
10/2003, Xtra Datenbank
- [24] <http://www.idsoftware.com/business/home/techdownloads/>
10/2003, 3D-Engine IDSoftware

-
- [25] <http://unreal.epicgames.com/>
10/2003, 3D-Engine Epicgames
- [26] <http://collective.valve-erc.com/index.php>
10/2003, 3D-Engine Valve
- [27] <http://www.3dengines.de/>
10/2003, 3D-Engines
- [28] <http://java.sun.com/products/java-media/3D/index.html>
10/2003, Java3D
- [29] <http://www.web3d.org>
10/2003, Extensible 3D
- [30] <http://www.blaxxun.de>
10/2003, Blaxxun
- [31] <http://www.parallelgraphics.com>
10/2003, Parallegraphics
- [32] <http://www.heise.de/newsticker/data/jk-04.11.03-001/>
10/2003, Marktstudie von Netcraft
- [33] <http://www.mysql.de>
10/2003, MySql Datenbank
- [34] <http://www.sybase.com/products/enterprisemodeling/powerdesigner>
10/2003, Powerdesigner
- [35] <http://www.apple.com/quicktime/qtvr/authoringstudio/index.html>
10/2003, Quicktime VR

-
- [36] <http://www.tecchannel.de/internet/760/0.html>
10/2003, X3D
- [37] <http://selfaktuell.teamone.de/>
10/2003, SelfHTML 8.0
- [38] <http://www.heise.de/newsticker/data/tol-23.04.03-002/>
10/2003, Studie zu DSL

Glossar

- ADSL** ADSL steht für Asymmetric Digital Subscriber Line und ist ein neuer Standard, der es erlaubt mehr Daten über herkömmliche Kupferleitungen zu schicken.
- API** API ist die Abkürzung für Application Programmers Interface. Es definiert die Schnittstelle zwischen Programmen und dem Betriebssystem. Diese Schnittstelle stellt Programmerroutinen bereit, mit denen andere Programme auf Hardwareroutinen zugreifen können.
- Browser** Ein Browser ist ein Programm, das den Zugriff und die Darstellung von Seiten des World Wide Web ermöglicht. Hauptsächlich sind Web-Browser dafür gedacht HTML-Dokumente und die dazugehörigen Bilder aus dem Internet herunterzuladen und anzuzeigen.
- CRT** CRT steht für Kathodenstrahlröhre (engl. Cathode-Ray Tube) und wird in herkömmlichen Monitoren und Fernsehgeräten eingebaut. Ein normales Fernsehgerät verfügt über eine Kathodenstrahlröhre mit einer speziellen Maske, durch die kleine Punkte oder Linien in den Farben rot, grün und blau ein farbiges Bild aufbauen.
- DirectX** DirectX ist eine von Microsoft entwickelte Programmierschnittstelle zur Entwicklung von 3D-Applikationen speziell für den Spielbereich.
- HTML** HTML steht für Hypertext Markup Language und ist die Beschreibungssprache der Bildschirmseiten im WWW des Internet, die mittlerweile in der Version 4.0 vorliegt. Zum Aufrufen und Lesen der HTML-Dokumente benötigt man ein Anwendungsprogramm, den WWW- oder Web-Browser.

ISDN	Integrated Services Digital Network. ISDN nutzt die bestehenden Telefonleitungen und überträgt alle Daten digital statt wie bisher analog
ISO	ISO ist die Abkürzung für "International Standardisation Organization" und ist eine internationale Organisation, die die Entwicklung weltweiter Normen - für alle Sachgebiete - koordiniert und für deren Veröffentlichung sorgt.
JDBC	JDBC ist die Abkürzung für Java Database Connectivity und ist eine Schnittstelle mit der man eine Verbindung zwischen einem Java-Programm und einer Datenbank herstellen kann.
LCD	LCD ist die Abkürzung von Liquid Crystal Display (Flüssigkristallanzeige). Die Flüssigkristalle können ihre Ausrichtung verändern und lassen so Licht durch bzw. nicht durch. Damit können verschiedene Bereiche des Bildschirms zum Leuchten gebracht werden. Dieses Prinzip wird zur Realisierung flacher Bildschirme, wie es etwa bei Notebooks oder Laptops eingesetzt.
ODBC	Unter "Open Database Connectivity" versteht man eine 1992 von der Firma Microsoft entwickelte Software-Schnittstelle, die den Zugriff aus einem Anwendungsprogramm auf unterschiedliche Datenbanken gewährleisten soll.
OpenGL	OpenGL ist eine von Silicon Graphics (SGI) entwickelte hardware-unabhängige Programmierschnittstellen zur Entwicklung von 3D-Applikationen.
Plugin	Ein Plugin ist ein Programm, dass sich als Teil eines anderen Programms einfügt, um zusätzliche Funktionen zur Verfügung zu stellen.

Danksagung

An dieser Stelle möchte ich mich bei meinen Betreuern für die Unterstützung bei meiner Diplomarbeit bedanken. Besonderer Dank gilt Herr Professor Jung, der den Kontakt zur Berliner Geschichtswerkstatt herstellte, sich Zeit für mich nahm und Anregungen zur Diplomarbeit gab. Von der Seite der Berliner Geschichtswerkstatt bedanke ich mich bei Frau Layer-Jung, die mir die notwendigen Informationsmaterialien zum Zwangsarbeiterlager Niederschöneweide zur Verfügung gestellt hat und Anregungen zur Diplomarbeit vom Standpunkt eines Historikers gab.

Eigenständigkeitserklärung

Hiermit versichere ich, dass ich die vorliegende Diplomarbeit selbständig und nur unter Verwendung der angegebenen Quellen und Hilfsmittel verfasst habe. Die Arbeit wurde bisher in gleicher oder ähnlicher Form keiner anderen Prüfungsbehörde vorgelegt.

Chee Fai Cheung

Berlin, 03.12.2003